Supplementary Material for Surrogate Benchmarks for Hyperparameter Optimization

Katharina Eggensperger and Frank Hutter
University of Freiburg
{eggenspk, fh}@cs.uni-freiburg.de

Holger H. Hoos and Kevin Leyton-Brown
University of British Columbia
{hoos, kevinlb}@cs.ubc.ca

A Data preprocessing

For each benchmark we studied for this paper, we preprocessed the data gathered by running the hyperparameter optimizers and random search as follows:

1. We extracted all available configuration/performance pairs from the runs. For benchmarks that used cross-validation, we encoded the cross-validation fold of each run as an additional categorical parameter (for benchmarks without cross validation, that parameter was set to a constant).
2. We removed entries with invalid results caused by algorithm crashes.
3. For data from benchmarks featuring conditional parameters, we replaced the values of inactive conditional parameters with a default value. We encoded categorical features using a one-hot (aka 1-in-k) encoding, which replaced any single categorical parameter λ with domain $\Lambda = \{k_1, \dots k_n\}$ by n binary parameters, only the i-th of which is true for data points where λ is set to k_i.

B Additional Tables and Figures

Due to limited space in the main paper we report full experimental results in the following.

Evaluation of Raw Model Performance

We report the 5-fold crossvalidated root mean squared error (RMSE) and Spearman's rank correlation coefficient (CC) for 8 algorithms on 9 datasets in Table B.1 (detailed version of Table 3 from the main paper).

Table **??** shows analogous results for a reduced set of 4 algorithms in the leave-one-optimizer-out (leave-ooo) setting. Figure B.1 (extended version of Figure 1 from the main paper) shows qualitative results for these leave-ooo experiments as scatterplots.

Using Surrogates as Hyperparameter Optimization Benchmarks

Figure B.3 (detailed version of Figure 2 from the main paper) shows the performance of the optimizer on the real benchmark and on surrogate-based benchmarks for all 9 optimization problems (where surrogates were trained in the leave-ooo setting).

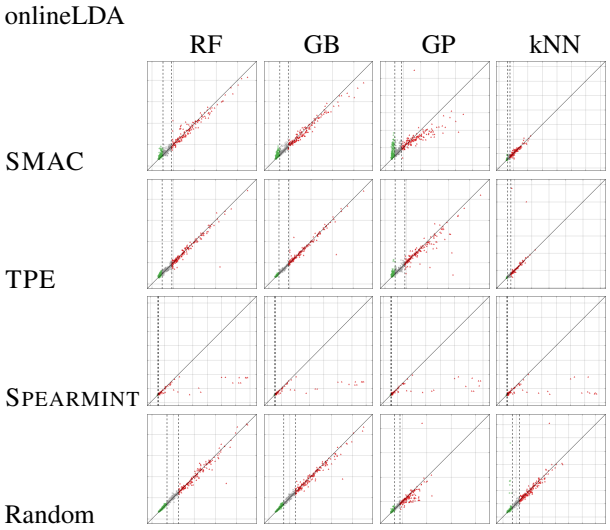
While the main paper only discusses surrogate benchmark results in the leave-ooo setting, here we report analogous results when using all data. Table B.3 offers a quantiative evaluation of these surrogates trained on all data (comparing the performance of SMAC, SPEARMINT, and TPE on the real benchmarks versus their performance on the surrogate benchmarks). Figure B.2 shows the performance of the optimizer on the real benchmark and on surrogate-based benchmarks for all 9 optimization problems (where surrogates were trained based on all data).

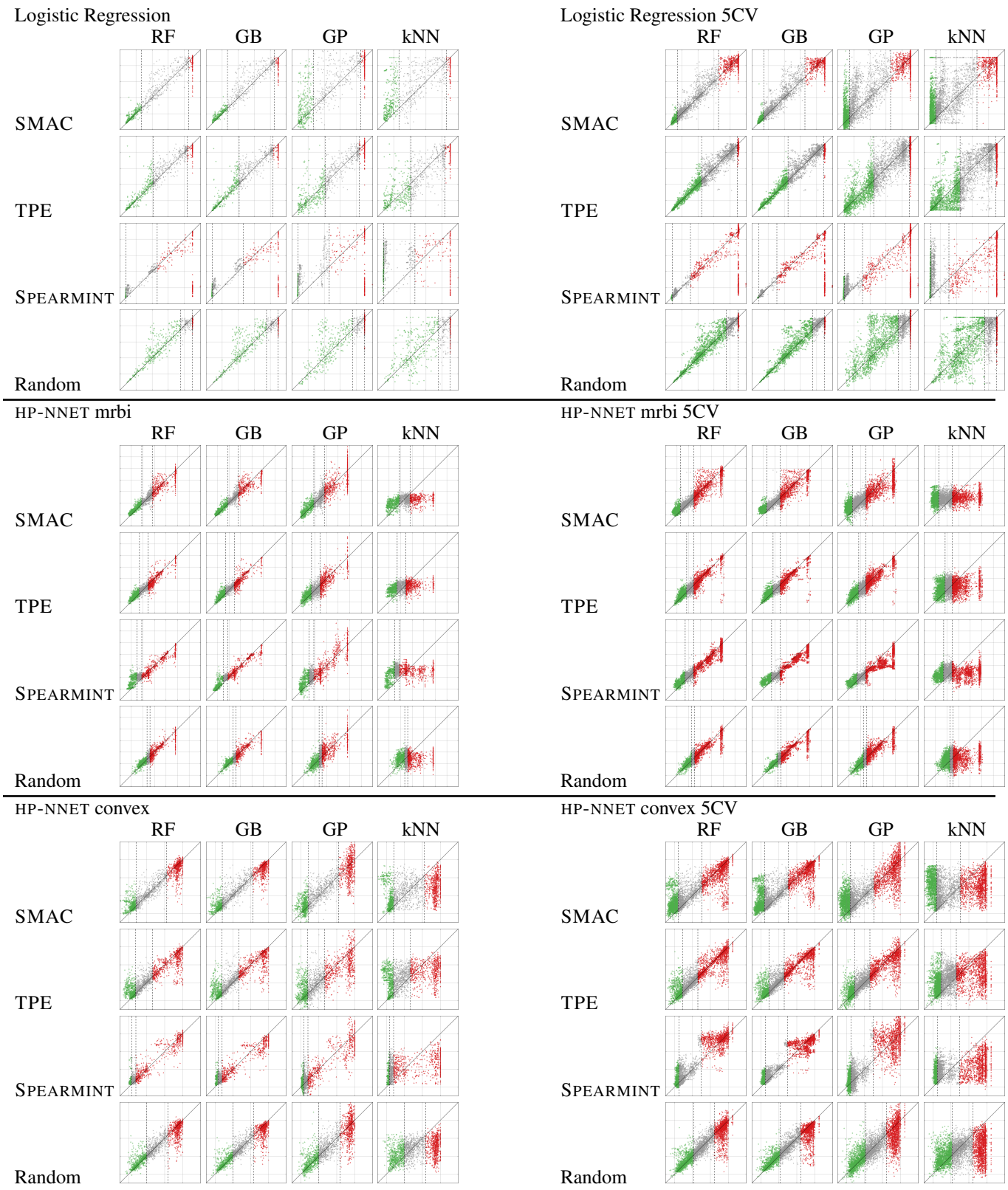
Table B.1: Regression performance in the **crossvalidation** setting. We report average RMSE and CC for a 5-fold cross validation for different regression models for 9 benchmark problem datasets. For each entry, bold face indicates the best performance on this dataset, and underlined values are not statistically significantly different from the best according to a paired t-test (with $p = 0.05$). For models marked with an * we reduced the trainingdata to a subset of 2000 data points per fold. This table is an extended version of Table 3 from the main paper.

	onlineLDA		Log.Reg		Log.Reg 5CV		HP-NNET conv.		HP-NNET 5CV conv.		HP-NNET mrbi		HP-NNET 5CV mrbi		HP-DBNET conv.		HP-DBNET mrbi	
Model	RMSE	CC	RMSE	CC	RMSE	CC	RMSE	CC	RMSE	CC	RMSE	CC	RMSE	CC	RMSE	CC	RMSE	CC
GB	**29.7**	0.99	**0.061**	0.95	**0.028**	0.96	**0.031**	0.95	0.026	0.96	**0.026**	0.96	0.019	0.98	**0.068**	0.91	**0.052**	0.92
RF	32.7	**0.99**	0.068	**0.95**	0.029	**0.98**	0.031	**0.95**	**0.025**	**0.97**	0.027	**0.96**	**0.018**	**0.98**	0.070	**0.91**	0.052	**0.92**
SVR	145.0	0.98	0.124	0.88	0.108	0.89	0.069	0.81	0.061	0.83	0.063	0.77	0.058	0.78	0.098	0.75	0.093	0.72
GP*	77.0	0.94	0.114	0.89	0.125	0.88	0.041	0.91	0.038	0.92	0.035	0.94	0.031	0.95	0.095	0.76	0.081	0.80
NuSVR*	144.9	0.98	0.131	0.86	0.141	0.86	0.065	0.85	0.066	0.83	0.053	0.88	0.055	0.86	0.108	0.68	0.089	0.73
KNN	154.4	0.97	0.146	0.88	0.137	0.89	0.085	0.64	0.085	0.62	0.091	0.55	0.086	0.61	0.106	0.67	0.106	0.62
Lin.Reg.	198.7	0.86	0.252	0.66	0.232	0.78	0.084	0.60	0.094	0.46	0.091	0.55	0.095	0.51	0.100	0.73	0.094	0.70
Rid.Reg.	198.7	0.86	0.252	0.66	0.232	0.78	0.084	0.60	0.094	0.46	0.091	0.55	0.095	0.51	0.100	0.73	0.094	0.70

Table B.2: Regression performance in the **leave-one-optimizer-out** setting. We report average RMSE and CC for 4 regression models for 9 benchmark problem datasets. Bold face indicates the best value across all regression models on this dataset. For models marked with an * we reduced the trainingdata to a subset of 2000 data points.

	onlineLDA		Log.Reg		Log.Reg 5CV		HP-NNET conv.		HP-NNET 5CV conv.		HP-NNET mrbi		HP-NNET 5CV mrbi		HP-DBNET conv.		HP-DBNET mrbi	
Model	RMSE	CC	RMSE	CC	RMSE	CC	RMSE	CC	RMSE	CC	RMSE	CC	RMSE	CC	RMSE	CC	RMSE	CC
GB	**100.0**	0.96	**0.123**	**0.86**	**0.095**	0.90	**0.037**	**0.87**	0.044	**0.86**	0.035	0.90	**0.035**	0.90	**0.078**	0.85	**0.059**	**0.87**
RF	101.9	**0.96**	0.130	0.84	0.098	**0.93**	0.039	0.85	**0.044**	0.86	**0.032**	**0.92**	0.036	**0.91**	0.081	**0.86**	0.060	0.87
GP*	136.0	0.89	0.183	0.73	0.158	0.76	0.053	0.79	0.056	0.81	0.052	0.82	0.050	0.86	0.103	0.62	0.088	0.73
KNN	190.6	0.92	0.228	0.68	0.198	0.72	0.093	0.46	0.103	0.31	0.099	0.34	0.103	0.24	0.118	0.51	0.126	0.26




Logistic Regression
RF
GB
GP
kNN
SMAC
TPE
SPEARMINT
Random
Logistic Regression 5CV
RF
GB
GP
kNN
SMAC
TPE
SPEARMINT
Random
HP-NNET mrbi
RF
GB
GP
kNN
SMAC
TPE
SPEARMINT
Random
HP-NNET mrbi 5CV
RF
GB
GP
kNN
SMAC
TPE
SPEARMINT
Random
HP-NNET convex
RF
GB
GP
kNN
SMAC
TPE
SPEARMINT
Random
HP-NNET convex 5CV
RF
GB
GP
kNN
SMAC
TPE
SPEARMINT
Random

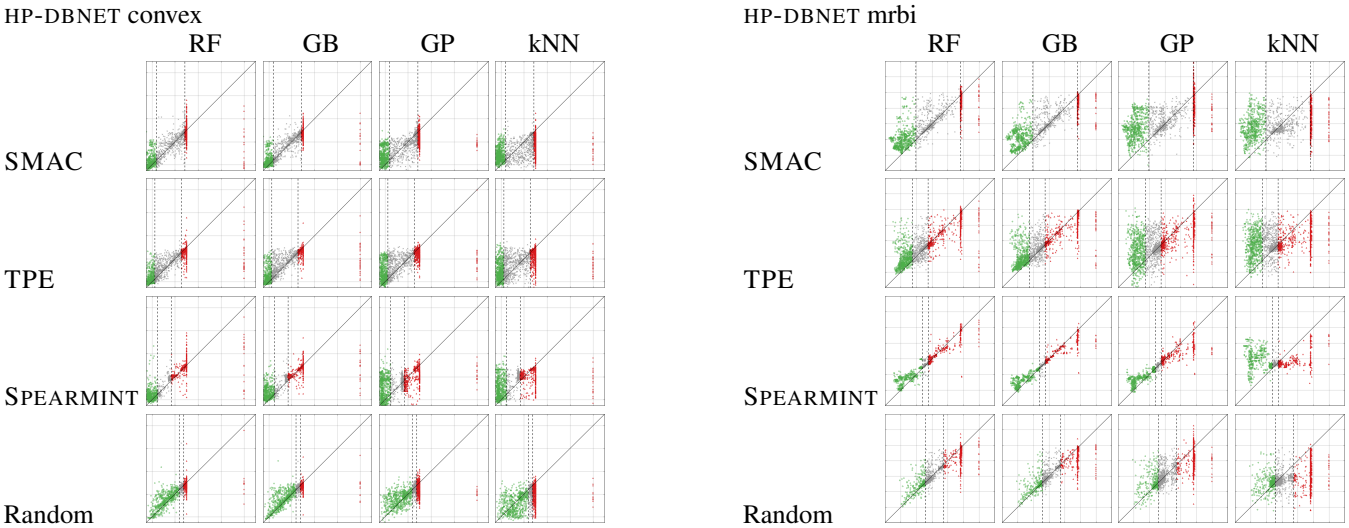
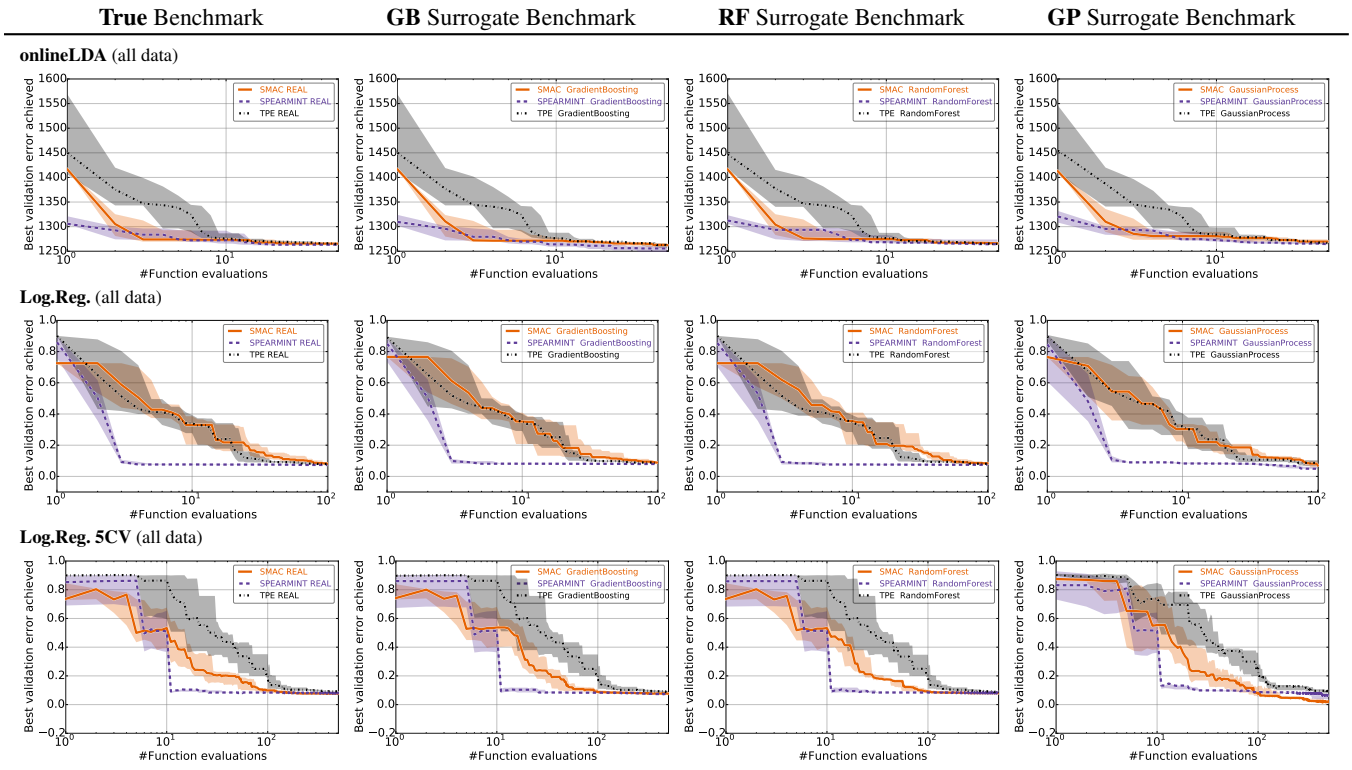


Figure B.1: True test performance (x-axis) vs. regression model predictions (y-axis). We report results for 4 regression models and 9 datasets. The models were trained on the leave-ooo and tested on the left out data. All plots in one row have the same axes. Each marker represents the performance of one configuration; Configurations on the diagonal are predicted perfectly, configurations below the diagonal were predicted to perform better than they really did, while configurations above the diagonal were predicted to perform worse than they really did. This figure shows the results on all datasets analogously to Figure 1 from the main paper.



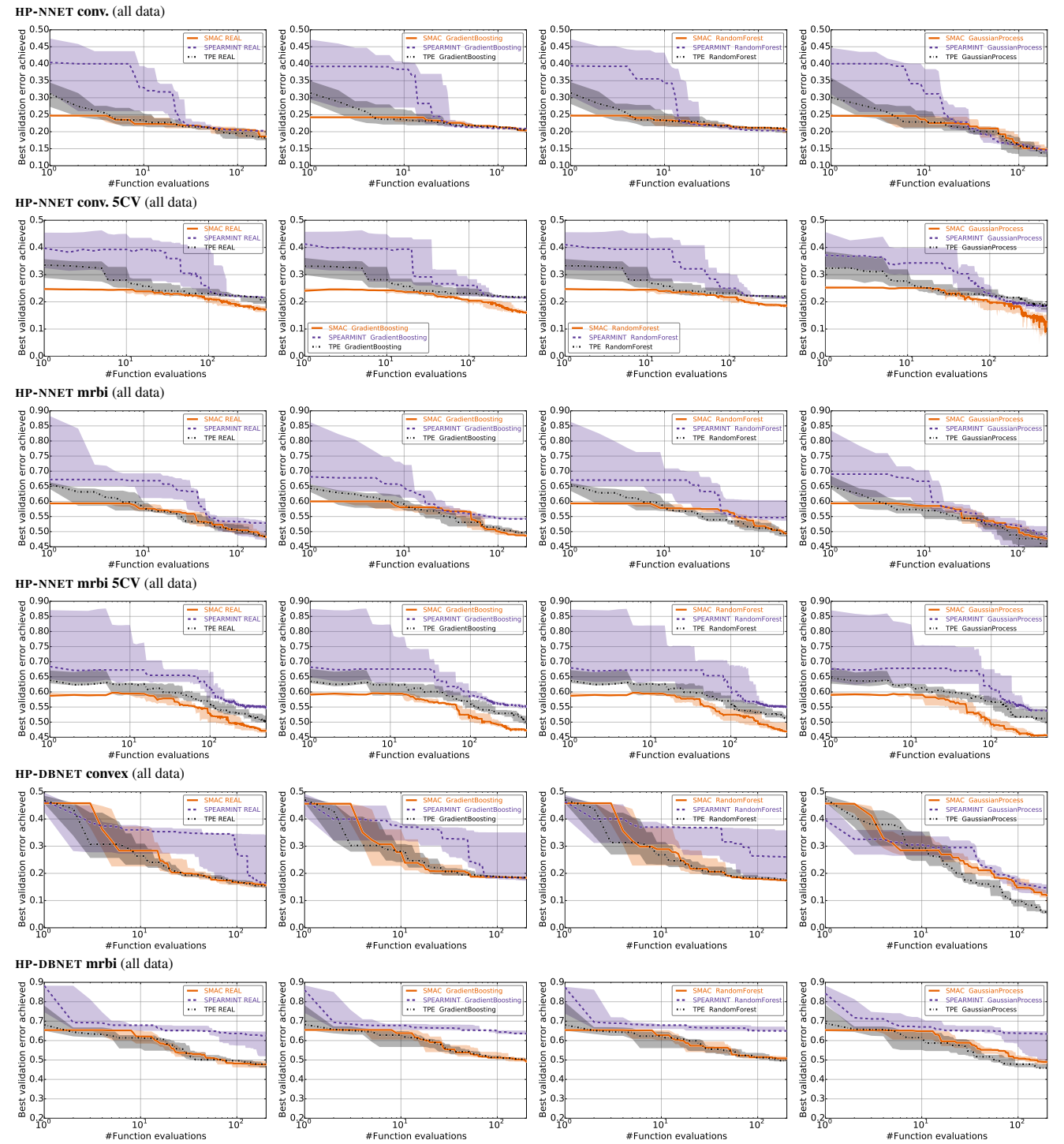
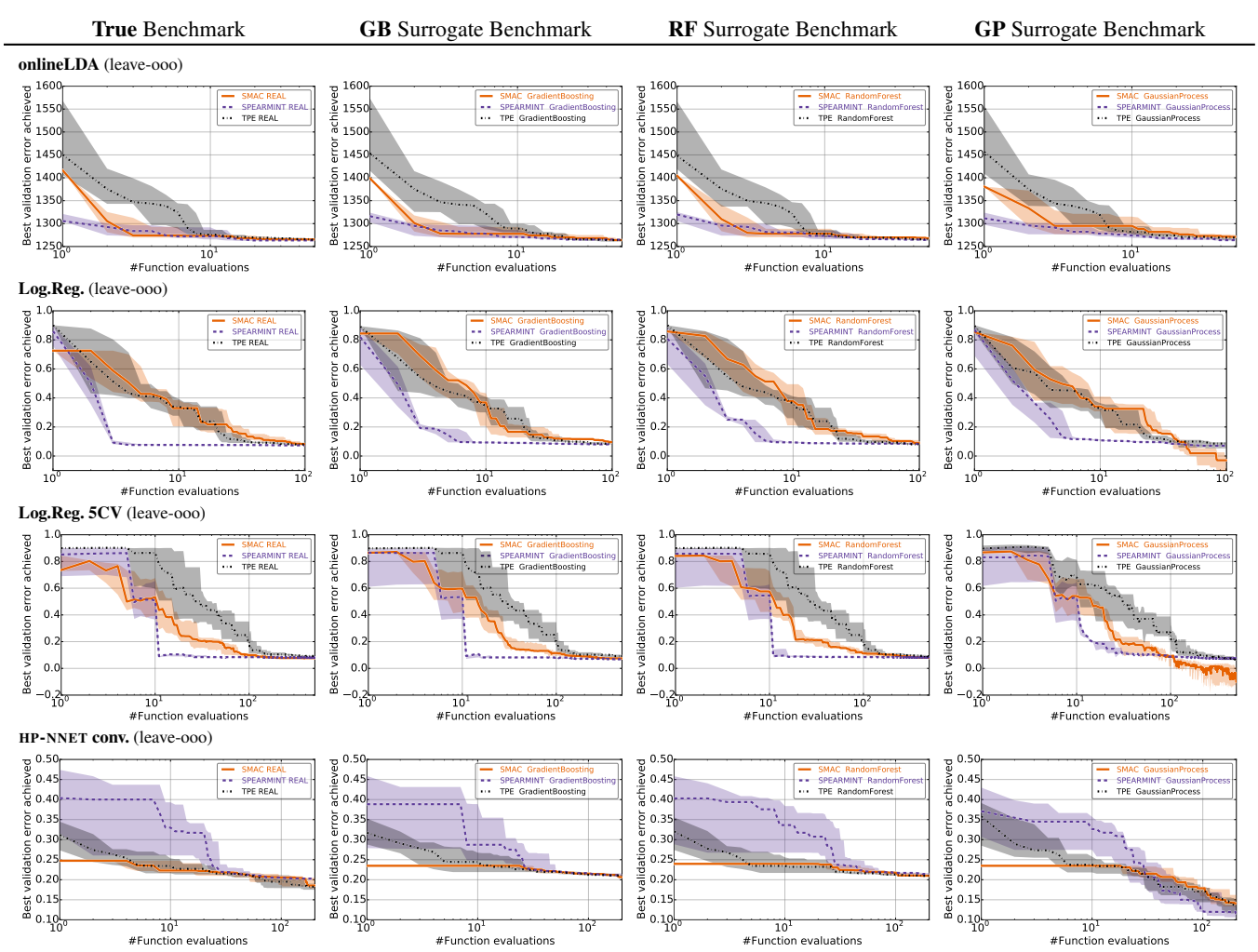


Figure B.2: Best performance found by different optimizers over time. We plot median and quartile of best performance across 10 runs of each optimizer over time on the real benchmark (left column) and on surrogates trained on **all data**.

		Results obtained on real benchmark			Results obtained on RF-based surrogate			Results obtained on GP-based surrogate		
		SMAC	**Spearmint**	**TPE**	**SMAC**	**Spearmint**	**TPE**	**SMAC**	**Spearmint**	**TPE**
Experiment	#evals	Valid. loss	Valid. loss	Valid. loss	Valid. loss	Valid. loss	Valid. loss	Valid. loss	Valid. loss	Valid. loss
Log.Reg.	100	0.08±0.00	**0.07**±0.00	0.08±0.00	0.09±0.02	**0.07**±0.00	0.08±0.00	0.07±0.02	**0.06**±0.01	0.09±0.02
onlineLDA	50	1266.4±4.4	1264.3±4.9	**1263.7**±3.0	1267.8±4.6	1269.0±7.6	**1265.0**±2.3	**1266.6**±16.3	1268.9±9.2	1267.9±3.0
HP-NNET convex	200	0.19±0.01	0.20±0.01	**0.19**±0.01	**0.20**±0.01	0.20±0.00	0.20±0.01	0.15±0.02	0.14±0.01	**0.13**±0.02
HP-NNET mrbi		0.49±0.01	0.51±0.03	**0.48**±0.01	0.50±0.02	0.56±0.07	**0.49**±0.01	0.48±0.02	0.48±0.05	**0.46**±0.03
HP-DBNET convex		0.15±0.01	0.23±0.10	**0.15**±0.01	**0.17**±0.01	0.26±0.09	0.18±0.02	0.11±0.03	0.14±0.03	**0.06**±0.01
HP-DBNET mrbi		**0.47**±0.02	0.59±0.08	0.47±0.02	0.50±0.02	0.65±0.02	**0.50**±0.02	0.49±0.03	0.58±0.01	**0.47**±0.02
Log.Reg 5CV	500	**0.08**±0.00	0.08±0.00	0.09±0.01	0.08±0.00	**0.08**±0.00	0.09±0.01	**0.03**±0.02	0.06±0.02	0.09±0.02
HP-NNET convex 5CV		**0.19**±0.01	0.23±0.05	0.21±0.01	**0.20**±0.01	0.23±0.05	0.21±0.01	**0.14**±0.05	0.18±0.02	0.18±0.03
HP-NNET mrbi 5CV		**0.48**±0.01	0.55±0.03	0.51±0.02	**0.49**±0.03	0.57±0.04	0.51±0.02	**0.47**±0.00	0.52±0.03	0.52±0.03

Table B.3: Losses obtained for all optimizers and benchmarks. We report the three result tables for the real benchmarks (left), RF-based surrogate benchmarks (middle), and GP-based surrogate benchmarks (right), where the surrogate models learned from **all available data**. We report rounded means and standard deviation across 10 runs of each optimizer. For each benchmark, bold face indicates the best mean loss, and underlined values are not statistically significantly different from the best according to an unpaired t-test (with p=0.05).



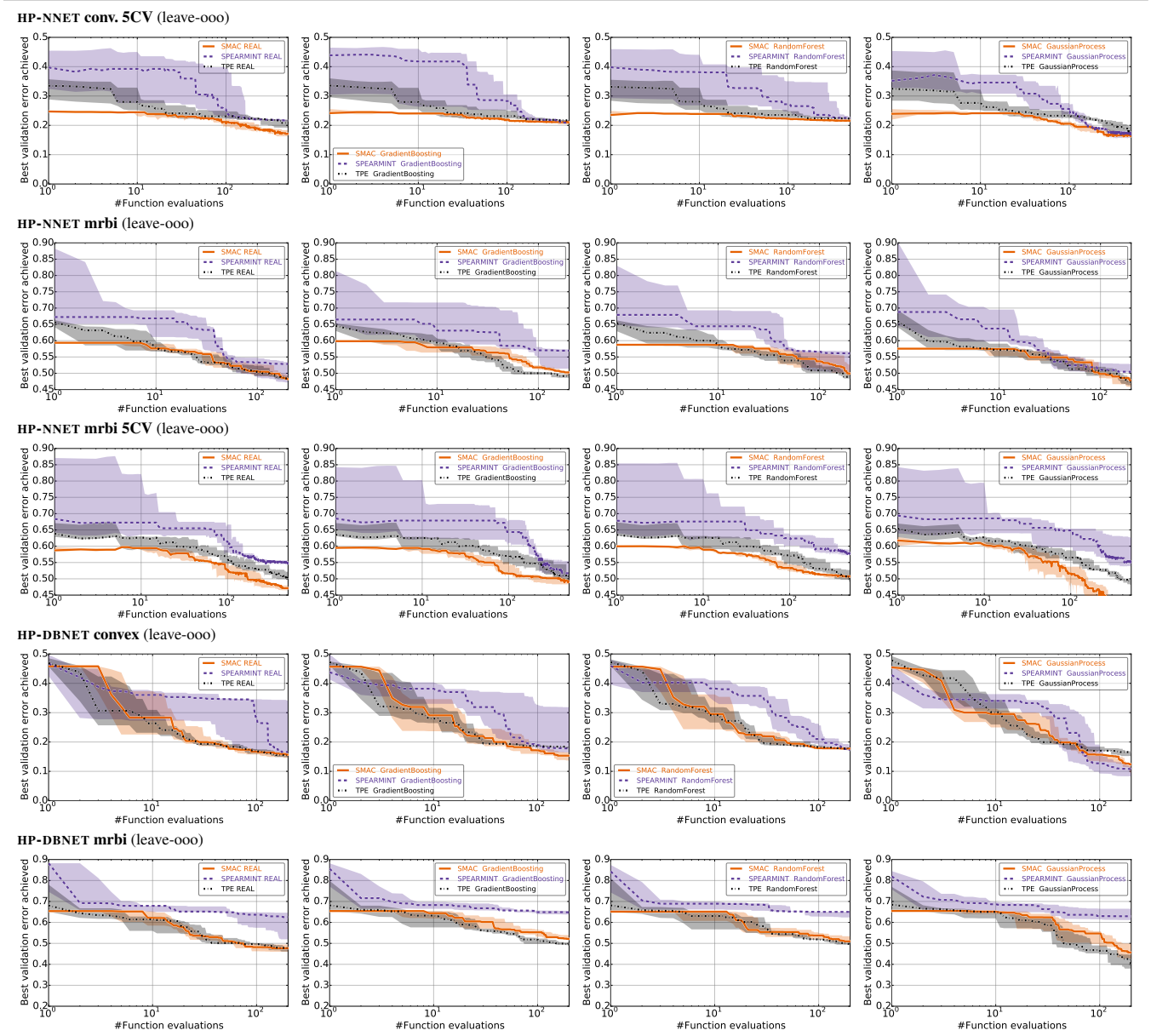


Figure B.3: Best performance found by different optimizers over time. We plot median and quartile of best performance across 10 runs of each optimizer over time on the real benchmark (left column) and on surrogates trained on **leave-ooo data**. This figure shows the results on all datasets analogously to Figure 2 from the main paper.

C Preliminary Results on Evaluating the Quality of a Surrogate

In order to evaluate the quality of a surrogate we could also evaluate the configurations selected on the surrogate during an optimization run. In this section we provide first results for such experiments for the logistic regression benchmark. We evaluated all configurations that were selected on the surrogate benchmark, which was trained on leave-ooo data. In Figure C.4 we compare the quality of the GP- and RF-based surrogate benchmarks.

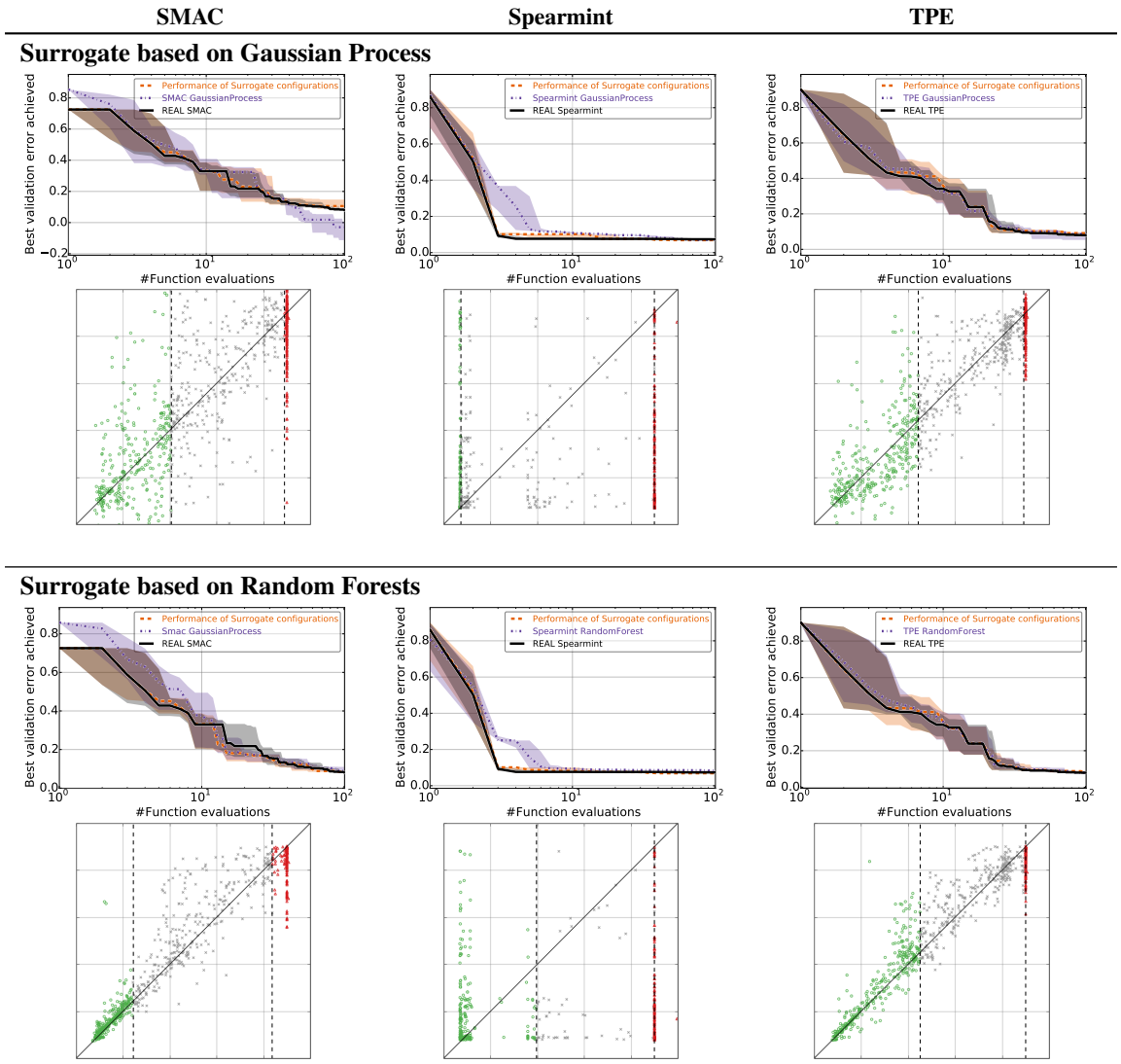


Figure C.4: The first row in each block shows the best performance found over time for the logistic regression benchmark. We plot median and quartile of best performance across 10 runs of each optimizer over time on the real benchmark and on the surrogate-based benchmark (which was trained on **leave-ooo data**). We also show the real performance (orange dashed line) of the optimization runs when evaluating the configurations selected on the surrogate benchmark on the real benchmark. The second row contains respective scatterplots (showing error rates ranging from 0 to 1) comparing predicted performance (y-axis) to true performance (x-axis) of these configurations. We highlight the best and worst third of the evaluated configurations.