Efficient Global Optimization with Indefinite Kernels

Technology Arts Sciences TH Köln

Martin Zaefferer, Thomas Bartz-Beielstein

TH Köln, Faculty of Computer Science and Engineering Science, `firstname.lastname@th-koeln.de`

SPOT SEVEN

Indefinite Kernels: Motivation

- Expensive optimization problems → surrogate model-based optimization
- Popular choice:
  - Kriging / Gaussian Processes (GP) → Efficient Global Optimization (EGO)
- Typical requirement: Definiteness of
  - Distances: Conditionally Negative Semi-Definite (CNSD)
  - Kernel (correlation): Positive Semi-Definite (PSD)
- Conflict:
  - Classical, definite kernels may be unavailable
  - New kernels, e.g., for combinatorial problems
  - Highly customized kernels (a priori, expert knowledge)
  - Definiteness unknown, or known to be lacking
- Some results exist: Support Vector Machines (SVM)
- Little research for GP, EGO

Question 1 - Availability of methods for indefinite kernels / distances?
Question 2 - Transfer, adapt to GP, EGO?
Question 3 - Performance of different methods?

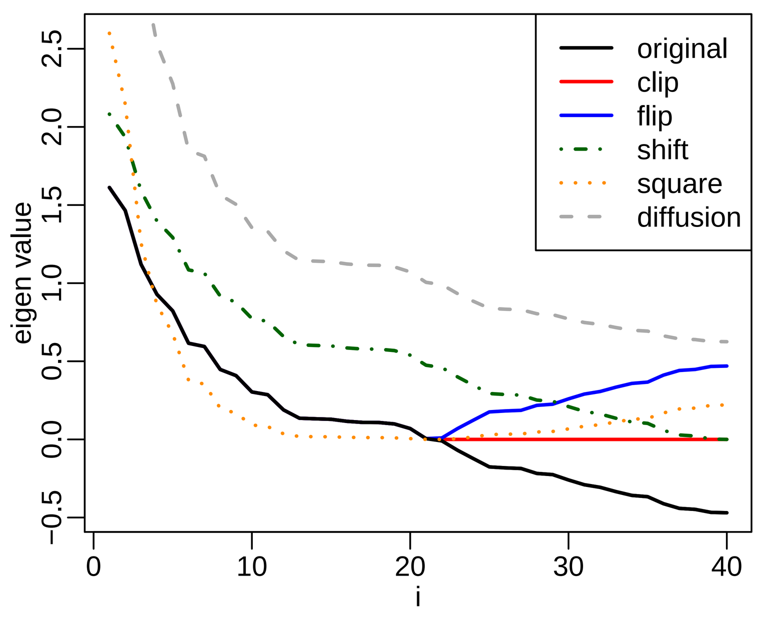
Methods

- **Spectrum transformations**

spectrum *clip*, *flip*, *shift*, *square*, *diffusion*

Apply to distance d (CNSD/NSD) or kernel k (PSD)

Additional condition repair:
$\mathrm{d}(x,x') \geq 0 \quad \mathrm{d}(x,x) = 0$
$-1 \leq \mathrm{k}(x,x') \leq 1 \quad \mathrm{k}(x,x) = 1$



- **Nearest CNSD / PSD matrix**
- **Feature Embedding**

Experiments

Benchmark Problems

- Artificial benchmarks, permutation problems, distance based
  - $\mathrm{d}_{Lp}(x,x') = (\sum_{i=1}^{n} |x_i - x'_i|^p)^{1/p}$ with $p = 1/2$ (non-metric)
  - interchange distance (metric for permutations, non-CNSD)
- Data type: permutations with $m = 5, 7, 10$ elements
- Varying ruggedness

EGO Settings

- Surrogate model: GP
- Kernel: Gaussian $\mathrm{k}(x,x') = e^{-\theta \mathrm{d}(x,x')}$ (same distance as in benchmark)
- Optimizer: GA - maximizing expected improvement

Quality Measures

- Prediction: RMSE - Uncertainty: Cramer von Mises (CVM) statistic
- Optimization performance: after 20 & 100 evaluations

Statistical test based ranks for: RMSE (R), CVM (C), best after 20 (F_a) & 100 (F_b) evaluations. P: percentage of problems solved.

names	R	C	F_a	F_b	P	names	R	C	F_a	F_b	P
T.flip.PSD	6	4	1	1	1	T.square.PSD.shift	6	5	2	2	0.91
F.clip.CNSD	1	6	1	1	0.99	standard.shift	7	4	2	2	0.91
F.clip.CNSD.shift	1	8	1	1	0.98	T.square.NSD	3	2	2	2	0.89
T.clip.NSD	1	2	1	1	0.96	T.square.CNSD	3	2	2	2	0.89
T.clip.CNSD	1	2	1	1	0.96	T.square.CNSD.shift	3	4	3	2	0.92
F.flip.CNSD.shift	1	10	1	1	0.95	F.square.PSD	4	4	2	3	0.86
near.CNSD	3	4	1	1	0.94	T.square.PSD	5	3	2	3	0.85
F.flip.CNSD	1	7	1	1	0.94	T.clip.PSD	4	3	2	3	0.84
T.flip.PSD.shift	3	5	2	1	0.98	F.clip.PSD	4	3	2	3	0.84
T.clip.CNSD.shift	2	3	2	1	0.98	standard	4	3	2	3	0.84
near.CNSD.shift	3	5	2	1	0.98	near.PSD	5	3	2	3	0.83
F.clip.PSD.shift	3	5	2	1	0.97	F.clip.NSD.shift	2	9	3	3	0.84
T.clip.PSD.shift	3	7	2	1	0.97	F.clip.NSD	2	9	3	3	0.84
T.clip.NSD.shift	2	3	2	1	0.97	F.flip.NSD.shift	4	8	3	3	0.83
F.flip.PSD	7	4	2	1	0.96	F.flip.NSD	4	6	3	3	0.82
F.flip.PSD.shift	6	4	2	1	0.95	near.PSD.shift	3	8	3	4	0.8
T.flip.NSD.shift	2	2	2	1	0.94	F.square.PSD.shift	3	6	3	4	0.76
feature	3	1	2	1	0.94	F.square.CNSD	4	8	3	5	0.7
T.flip.NSD	1	1	1	2	0.92	F.square.CNSD.shift	3	9	4	5	0.73
feature.shift	3	2	3	1	0.94	F.square.NSD.shift	2	9	4	5	0.69
T.flip.CNSD.shift	2	2	2	2	0.93	F.square.NSD	4	10	3	6	0.67
T.square.NSD.shift	3	4	2	2	0.92	random			5	7	0.35
T.flip.CNSD	1	1	2	2	0.92						

Main effects for spectrum transformation - Intercept: F.square.CNSD

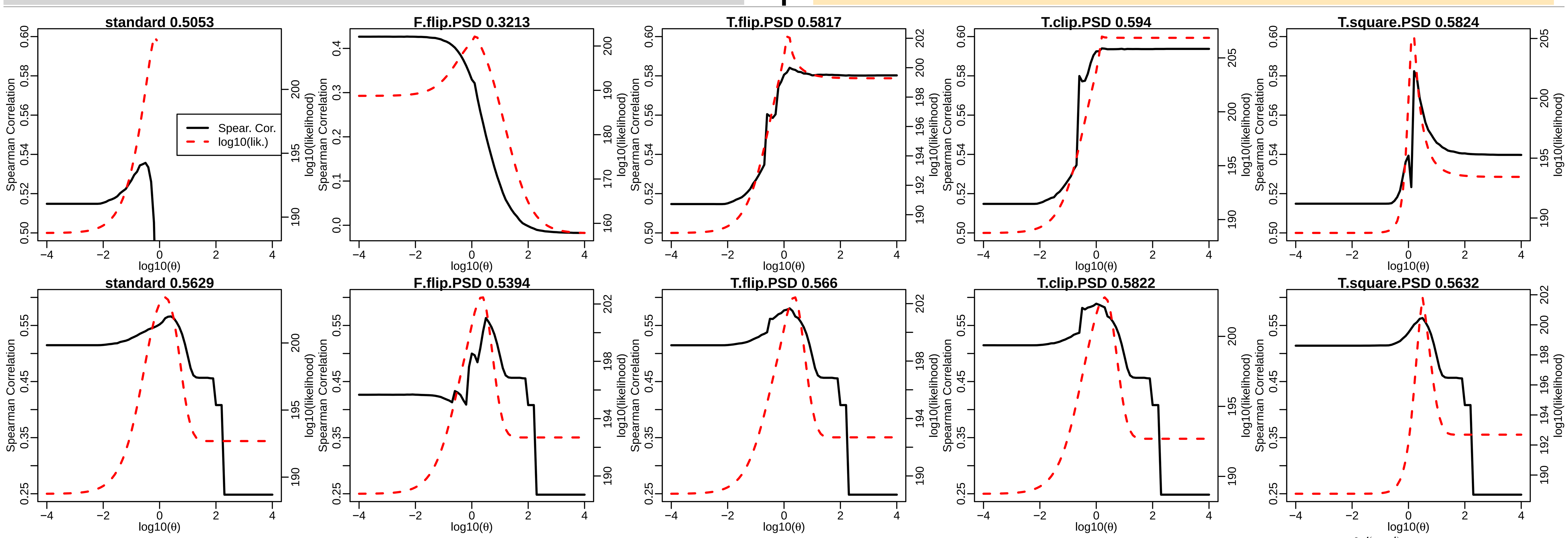
	F_b (rank after 100 eval.)			F_a (rank after 20 eval.)		
	Estimate	Std. Error	p-value	Estimate	Std. Error	p-value
Intercept	494.3276	2.8163	$<$ 2e-16	469.3359	5.4252	$<$ 2e-16
repair: T	-37.2445	2.1290	$<$ 2e-16	-30.7094	4.1011	7.44e-14
method: clip	-49.6153	2.6074	$<$ 2e-16	-77.4150	5.0228	$<$ 2e-16
method: flip	-52.1515	2.6074	$<$ 2e-16	-68.1126	5.0228	$<$ 2e-16
type: PSD	0.0664	2.6074	0.98	15.7471	5.0228	0.0017
type: NSD	23.5449	2.6074	$<$ 2e-16	55.7815	5.0228	$<$ 2e-16
shift	-9.2891	2.1290	1.29e-05	29.4767	4.1011	6.96e-13

Read, e.g., the effect of `repair: T` is negative, hence `repair: T` is better than the intercept (`repair: F`). Red cells: change of effect sign.

Conclusions and Outlook

- Standard approach works
- Definiteness correction mostly advantageous
- Spectrum diffusion fails, spectrum square performs rather poorly
- Nearest matrix approaches: not bad, but expensive
- Robust, good results: spectrum flip and clip
- Condition repair often beneficial

- More extensive experiments
- Efficient handling of new data with condition repair?
- Method selection?
- Theoretical interpretation?



Likelihood vs. Spearman Correlation between prediction and true values. First row: $\mathrm{k}_{nd}(x,x') = 1 - \theta \mathrm{d}(x,x')$. Second row: $\mathrm{k}_{gauss}(x,x') = e^{-\theta \mathrm{d}(x,x')}$. For the sake of visual comparability, these instances use a constant spectrum shift of $\eta = 10^{-4}$, to avoid failures in case of $\mathrm{k}_{nd}(x,x')$. Data base: 50 permutations ($m = 6$ elements) for training, 600 for testing, the distance function $\mathrm{d}(x,x')$ is interchange distance. Larger values are better. Number in headings: Correlation for max. likelihood model.

PPSN 2016, 14th International Conference on Parallel Problem Solving from Nature, September 2016, Edinburgh, UK