Accelerated Bayesian Optimisation through Weight-Prior Tuning - Supplementary

January 23, 2020

1 p-Norm Support Vector Machines

A canonical application of m-kernels is the p-norm support vector machine (SVM) (ℓ^p-SVM [3], max-margin L^p moment classifier [1]). Let $\mathtt{D} = \{(\mathbf{x}_i, y_i) \in \mathbb{X} \times \mathbb{Y}\}$ be a training set and $\boldsymbol{\varphi} : \mathbb{X} \to \mathbb{R}^d$ a feature map implied by a $2q$-kernel K, where $q \in \mathbb{N}\backslash\{0\}$. Following [3] the aim is to find a sparse (in $\mathbf{w} \in \mathbb{R}^d$) trained machine:

$$g\left(\mathbf{x}\right) = \mathbf{w}^{\mathrm{T}} \boldsymbol{\varphi}\left(\mathbf{x}\right) + b \tag{1}$$

to fit the data. The paramters $\mathbf{w} \in \mathbb{R}^d$, $b \in \mathbb{R}$ are found by solving the p-norm training problem, where $p \in \mathbb{R}$, $1 < p \leq 2$ is dual to $2q \in 2\mathbb{Z}_+$ (i.e. $\frac{1}{p} + \frac{1}{2q} = 1$):

$$\min_{\mathbf{w},b} R_p\left(\mathbf{w}, b, \boldsymbol{\xi}\right) = r(\tfrac{1}{p}\|\mathbf{w}\|_p^p) + \tfrac{C}{N}\textstyle\sum_i E(y_i, g(\mathbf{x}_i)) \tag{2}$$

where r is strictly monotonically increasing, E is an arbitrary empirical risk function, and the use of p-norm regularization with $1 < p \leq 2$ encourages sparsity in $\mathbf{w}$. Following [3, 2] it may be shown that:

$$\mathbf{w} = \textstyle\sum_{i_1,i_2,\ldots,i_{2q-1}} \alpha_{i_1}\alpha_{i_2}\ldots\alpha_{i_{2q-1}} \boldsymbol{\varphi}(\mathbf{x}_{i_1}) \odot \boldsymbol{\varphi}(\mathbf{x}_{i_2}) \odot \ldots \odot \boldsymbol{\varphi}(\mathbf{x}_{i_{2q-1}}) \tag{3}$$

(representor theorem) and hence:

$$g(\mathbf{x}) = \textstyle\sum_{i_1,i_2,\ldots,i_{2q-1}} \alpha_{i_1}\alpha_{i_2}\ldots\alpha_{i_{2q-1}} K(\mathbf{x}, \mathbf{x}_{i_1}, \mathbf{x}_{i_2}, \ldots, \mathbf{x}_{i_{2q-1}}) + b \tag{4}$$

Moreover we may completely suppress $\mathbf{w}$ (the m-kernel trick) and construct a dual training problem entirely in terms of $\boldsymbol{\alpha}$ [2, 3] - e.g. if $\mathbb{R} = \mathbb{Y}$, $E(y, g) = \frac{1}{2}(y - g)^2$ (ridge regression), the dual training problem is:

$$\begin{aligned} &\min_{\alpha} \ \tfrac{1}{2q} \sum_{i_0,i_1,\ldots,i_{2q-1}} \alpha_{i_0}\alpha_{i_1}\ldots\alpha_{i_{2q-1}} K_{i_0,i_1,\ldots,i_{2q-1}} + \tfrac{N}{2C}\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\alpha} - \mathbf{y}^{\mathrm{T}}\boldsymbol{\alpha} \\ &\text{such that:} \quad \textstyle\sum_i \alpha_i = 0 \end{aligned} \tag{5}$$

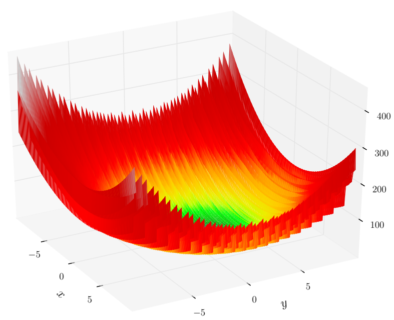
where $K_{i_0,i_1,\ldots,i_{2q-1}} = K(\mathbf{x}_{i_0}, \mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_{2q-1}})$. Similar results, analogous to the "standard" SVMs (e.g. binary classification) may be likewise constructed [2, 3, 1].

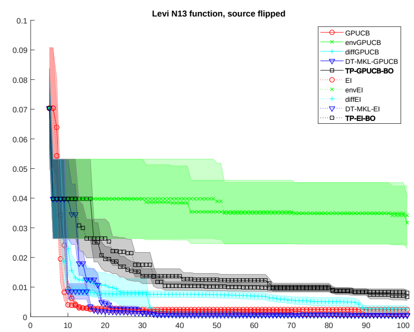
2 Counterexamples

The following functions are the counter-examples referred to in section 7, and represent special cases where our method will fail. See section 7 of the body for a full discussion.

2.1 Levi N.13 Function

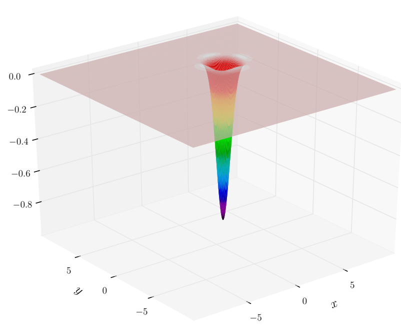
The Levi N.13 test function varies significantly on a very short lengthscale, so in this case the relatively small auxilliary dataset $\mathtt{A}$ cannot properly characterise the covariance structure, resulting in slowed convergence.

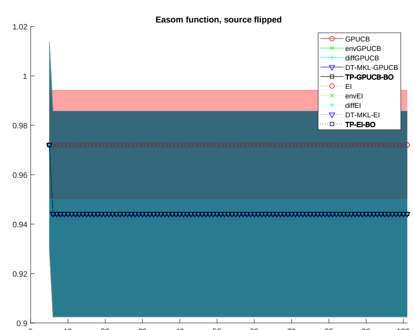




2.2 Easom Function

The Easom function is mostly flat (featureless), so the (random) sampling in the auxiliary dataset is consistent with a constant function, resulting in a trivial tuned covariance $K_2^{\mathtt{A}}(\mathbf{x}, \mathbf{x}') = 0$. As noted in section 7 of the body this is trivial to detect, and moreover none of the methods tested make any headway on this objective.





References

[1] R. Der and D. Lee. Large-margin classification in banach spaces. In *Proceedings of the JMLR Workshop and Conference 2: AISTATS2007*, pages 91–98, 2007.

[2] S. Salzo and J. A. K. Suykens. Generalized support vector regression: duality and tensor-kernel representation. *arXiv preprint arXiv:1603.05876*, 2016.

[3] S. Salzo, J. A. K. Suykens, and L. Rosasco. Solving ℓ^p-norm regularization with tensor kernels. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS) 2018*, April 2018.

[4] Wikipedia contributors. Test functions for optimization — Wikipedia, the free encyclopedia, 2019. [Online; accessed 3-October-2019].