Optimization as Estimation with Gaussian Processes in Bandit Settings (Supplement)

Zi Wang
MIT CSAIL

Bolei Zhou
MIT CSAIL

Stefanie Jegelka
MIT CSAIL

In this supplement, we provide proofs for all theorems and lemmas in the main paper, more exhaustive experimental results and details on the experiments.

1 Proofs

1.1 Proofs from Section 2

Lemma 2.1. *In any round t, the point selected by EST is the same as the point selected by a variant of GP-UCB with $\lambda_t = \min_{\boldsymbol{x}\in\mathfrak{X}} \frac{\hat{m}_t - \mu_{t-1}(\boldsymbol{x})}{\sigma_{t-1}(\boldsymbol{x})}$. Conversely, the candidate selected by GP-UCB is the same as the candidate selected by a variant of EST with $\hat{m}_t = \max_{\boldsymbol{x}\in\mathfrak{X}} \mu_{t-1}(\boldsymbol{x}) + \lambda_t\sigma_{t-1}(\boldsymbol{x})$.*

Proof. We omit the subscripts t for simplicity. Let $\boldsymbol{a}$ be the point selected by GP-UCB, and $\boldsymbol{b}$ selected by EST. Without loss of generality, we assume $\boldsymbol{a}$ and $\boldsymbol{b}$ are unique. With $\lambda = \min_{\boldsymbol{x}\in\mathfrak{X}} \frac{\hat{m} - \mu(\boldsymbol{x})}{\sigma(\boldsymbol{x})}$, GP-UCB chooses to evaluate

$$\boldsymbol{a} = \arg\max_{\boldsymbol{x}\in\mathfrak{X}} \mu(\boldsymbol{x}) + \lambda\sigma(\boldsymbol{x}) = \arg\min_{\boldsymbol{x}\in\mathfrak{X}} \frac{\hat{m} - \mu(\boldsymbol{x})}{\sigma(\boldsymbol{x})}.$$

This is because

$$\hat{m} = \max_{\boldsymbol{x}\in\mathfrak{X}} \mu(\boldsymbol{x}) + \lambda\sigma(\boldsymbol{x}) = \mu(\boldsymbol{a}) + \lambda\sigma(\boldsymbol{a}).$$

By definition of $\boldsymbol{b}$, for all $\boldsymbol{x} \in \mathfrak{X}$, we have

$$\begin{aligned}\frac{\Pr[M_{\boldsymbol{b}}|\hat{m}, \mathfrak{D}]}{\Pr[M_{\boldsymbol{x}}|\hat{m}, \mathfrak{D}]} &\approx \frac{Q(\frac{\hat{m}-\mu(\boldsymbol{b})}{\sigma(\boldsymbol{b})}) \prod_{\boldsymbol{x}'\neq\boldsymbol{b}} \Phi(\frac{\hat{m}-\mu(\boldsymbol{x}')}{\sigma(\boldsymbol{x}')})}{Q(\frac{\hat{m}-\mu(\boldsymbol{x})}{\sigma(\boldsymbol{x})}) \prod_{\boldsymbol{x}'\neq\boldsymbol{x}} \Phi(\frac{\hat{m}-\mu(\boldsymbol{x}')}{\sigma(\boldsymbol{x}')})} \\ &= \frac{Q(\frac{\hat{m}-\mu(\boldsymbol{b})}{\sigma(\boldsymbol{b})})\Phi(\frac{\hat{m}-\mu(\boldsymbol{x})}{\sigma(\boldsymbol{x})})}{Q(\frac{\hat{m}-\mu(\boldsymbol{x})}{\sigma(\boldsymbol{x})})\Phi(\frac{\hat{m}-\mu(\boldsymbol{b})}{\sigma(\boldsymbol{b})})} \\ &\geq 1.\end{aligned}$$

The inequality holds if and only if $\frac{\hat{m}-\mu(\boldsymbol{b})}{\sigma(\boldsymbol{b})} \leq \frac{\hat{m}-\mu(\boldsymbol{x})}{\sigma(\boldsymbol{x})}$ for all $\boldsymbol{x} \in \mathfrak{X}$, including $\boldsymbol{a}$, and hence

$$\frac{\hat{m} - \mu(\boldsymbol{b})}{\sigma(\boldsymbol{b})} \leq \frac{\hat{m} - \mu(\boldsymbol{a})}{\sigma(\boldsymbol{a})} = \lambda = \min_{\boldsymbol{x}\in\mathfrak{X}} \frac{\hat{m} - \mu(\boldsymbol{x})}{\sigma(\boldsymbol{x})},$$

which, with uniqueness, implies that $\boldsymbol{a} = \boldsymbol{b}$ and GP-UCB and EST select the same point.

For the other direction, we denote the candidate selected by GP-UCB by

$$\boldsymbol{a} = \arg\max_{\boldsymbol{x}\in\mathfrak{X}} \mu(\boldsymbol{x}) + \lambda\sigma(\boldsymbol{x}).$$

The variant of EST with $\hat{m} = \max_{\boldsymbol{x}\in\mathfrak{X}} \mu(\boldsymbol{x}) + \lambda\sigma(\boldsymbol{x})$ selects

$$\boldsymbol{b} = \arg\max_{\boldsymbol{x}\in\mathfrak{X}} \Pr[M_{\boldsymbol{x}}|\hat{m}, \mathfrak{D}].$$

We know that for all $\boldsymbol{x} \in \mathfrak{X}$, we have $\frac{\hat{m}-\mu(\boldsymbol{b})}{\sigma(\boldsymbol{b})} \leq \frac{\hat{m}-\mu(\boldsymbol{x})}{\sigma(\boldsymbol{x})}$ and hence $\hat{m} \leq \mu(\boldsymbol{b}) + \frac{\hat{m}-\mu(\boldsymbol{x})}{\sigma(\boldsymbol{x})}\sigma(\boldsymbol{b})$. Since $\hat{m} = \mu(\boldsymbol{a}) + \lambda\sigma(\boldsymbol{a})$, letting $\boldsymbol{x} = \boldsymbol{a}$ implies that

$$\hat{m} = \max_{\boldsymbol{x}\in\mathfrak{X}} \mu(\boldsymbol{x}) + \lambda\sigma(\boldsymbol{x}) \leq \mu(\boldsymbol{b}) + \lambda\sigma(\boldsymbol{b}).$$

Hence, by uniqueness it must be that $\boldsymbol{a} = \boldsymbol{b}$ and GP-UCB and EST select the same candidate. □

1.2 Proofs from Section 3

Lemma 3.2. *Pick $\delta \in (0, 1)$ and set $\zeta_t = (2\log(\frac{\pi_t}{2\delta}))^{\frac{1}{2}}$, where $\sum_{t=1}^{T} \pi_t^{-1} \leq 1$, $\pi_t > 0$. Then, for EST, it holds that $\Pr[\mu_{t-1}(\boldsymbol{x}_t) - f(\boldsymbol{x}_t) \leq \zeta_t\sigma_{t-1}(\boldsymbol{x}_t)] \geq 1 - \delta$, for all $t \in [1, T]$.*

Proof. Let $z_t = \frac{\mu_{t-1}(x_t) - f(x_t)}{\sigma_{t-1}(x_t)} \sim \mathcal{N}(0, 1)$. It holds that

$$\begin{aligned}\Pr[z_t > \zeta_t] &= \int_{\zeta_t}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-z^2/2}\,\mathrm{d}z \\ &= \int_{\zeta_t}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-(z-\zeta_t)^2/2 - \zeta_t^2/2 - z\zeta_t}\,\mathrm{d}z \\ &\leq e^{-\zeta_t^2/2} \int_{\zeta_t}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-(z-\zeta_t)^2/2}\,\mathrm{d}z \\ &= \frac{1}{2} e^{-\zeta_t^2/2}.\end{aligned}$$

A union bound extends this bound to all rounds:

$$\Pr[z_t > \zeta_t \text{ for some } t \in [1, T]] \leq \sum_{t=1}^{T} \frac{1}{2} e^{-\zeta_t^2/2}.$$

With $\zeta_t = (2\log(\frac{\pi_t}{2\delta}))^{\frac{1}{2}}$ and $\sum_{t=1}^{T} \pi_t^{-1} = 1$, this implies that with probability at least $1-\delta$, it holds that $\mu_{t-1}(\boldsymbol{x}_t) - f(\boldsymbol{x}_t) \leq \zeta_t \sigma_{t-1}(\boldsymbol{x}_t)$ for all $t \in [1, T]$. One may set $\pi_t = \frac{1}{6}\pi^2 t^2$, or $\pi_t = T$, in which case $\zeta_t = \zeta = (2\log(\frac{T}{2\delta}))^{\frac{1}{2}}$. □

Lemma 3.3. *If $\mu_{t-1}(\boldsymbol{x}_t) - f(\boldsymbol{x}_t) \leq \zeta_t \sigma_{t-1}(\boldsymbol{x}_t)$, the regret at time step t is upper bounded as $\tilde{r}_t \leq (\nu_t + \zeta_t)\sigma_{t-1}(\boldsymbol{x}_t)$, where $\nu_t \triangleq \min_{\boldsymbol{x}\in\mathfrak{X}} \frac{\hat{m}_t - \mu_{t-1}(\boldsymbol{x})}{\sigma_{t-1}(\boldsymbol{x})}$, and $\hat{m}_t \geq \max_{\boldsymbol{x}\in\mathfrak{X}} f(\boldsymbol{x})$, $\forall t \in [1, T]$.*

Proof. At time step $t \geq 1$, we have

$$\begin{aligned}
\tilde{r}_t &= \max_{\boldsymbol{x}\in\mathfrak{X}} f(\boldsymbol{x}) - f(\boldsymbol{x}_t) \\
&\leq \hat{m}_t - f(\boldsymbol{x}_t) \\
&\leq \hat{m}_t - \mu_{t-1}(\boldsymbol{x}_t) + \zeta_t \sigma_{t-1}(\boldsymbol{x}_t) \\
&= (\nu_t + \zeta_t)\sigma_{t-1}(\boldsymbol{x}_t).
\end{aligned}$$

□

2 Experiments

2.1 Approximate m

In the paper, we estimate m to be

$$\hat{m} = m_0 + \int_{m_0}^{\infty} 1 - \prod_{\boldsymbol{x}\in\mathfrak{W}} \Phi\Big(\frac{w - \mu(\boldsymbol{x})}{\sigma(\boldsymbol{x})}\Big)\,\mathrm{d}w \qquad (1)$$

which involves an integration from the current maximum m_0 of the observed data to positive infinity. In fact the factor inside the integration quickly approaches zero in practice. We plot $g(w) = 1 - \prod_{\boldsymbol{x}\in\mathfrak{W}} \Phi\Big(\frac{w-\mu(\boldsymbol{x})}{\sigma(\boldsymbol{x})}\Big)$ in Figure 1, which looks like half of a Gaussian distribution. So instead of numerical integration (which can be done efficiently), heuristically we can sample two values of $g(w)$ to fit $\hat{g}(w) = ae^{-(w-m_0)^2/2b^2}$ and do the integration $\int_{m_0}^{\infty} \hat{g}(w)\,\mathrm{d}w = \sqrt{2\pi}ab$ analytically to be more efficient. This method is what we called ESTa in the paper, while the numerical integration is called ESTn.

We notice that our estimation $\hat{m}$ can serve as a tight upper bound on the real value of the max of the function in practice. One example of is shown in Figure 1 with a 1-D GP function. This example shows how PI, ESTa and ESTn estimate m. Both ESTa and ESTn are upper bounds of the true maximum of the function, and ESTn is actually very tight. For PI, $\theta = \arg\max_{1\leq\tau<t} y_\tau + \epsilon$ is always a lower bound of an ϵ shift over the true maximum of the function.

2.2 Synthetic data

We show the examples of the functions we sampled from GP in Figure 2. The covariance function of GP is an isotropic Matérn kernel with parameters $\ell = 0.1, \sigma_f = 1$. The mean function is a linear function with a fixed random slope for different dimensions, and the constant is 1.

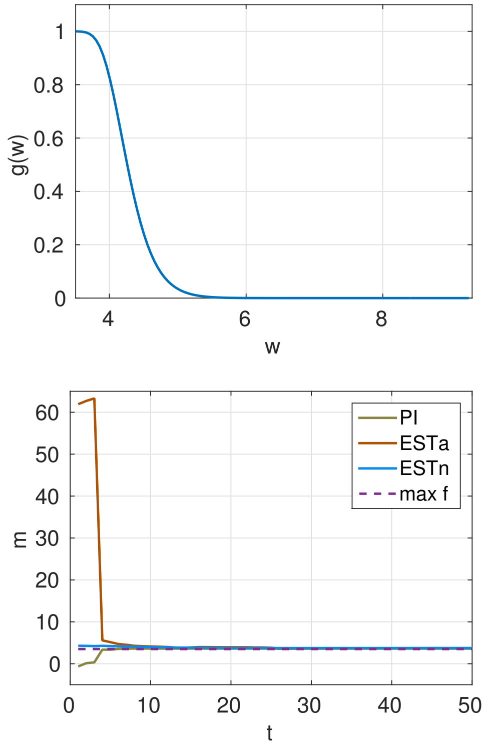

Figure 1: Top: $g(w)$, $w \in [m_0, +\infty)$; Bottom: estimation of m.

2.3 Initialization tuning for trajectory optimization

The 8 configurations of start state are $[7\ 1\ 0\ 0]$, $[7\ 0\ 0\ 0]$, $[1\ 0\ 0\ 0]$, $[1\ 1\ 0\ 0]$, $[2\ 0\ 0\ 0]$, $[2\ 1\ 0\ 0]$, $[3\ 0\ 0\ 0]$, $[3\ 1\ 0\ 0]$, where the first two dimensions denote the position and the last two dimension denote the speed. We only tune the first two dimension and keep the speed to be 0 for both directions. The target state is fixed to be $[5\ 9\ 0\ 0]$.

We can initialize the trajectory by setting the mid point of trajectory to be any point falling on the grid of the space (both x axis and y axis have range $[-2, 12]$). Then use SNOPT to solve the trajectory optimization problem, which involves an objective cost function (we take the negative cost to be a reward function to maximize), dynamics constraints, and obstacle constraints etc. Details of trajectory optimization are available in [11, 10].

We used the same settings of parameters for GP as in Section 2.2 for all the methods we tested and did kernel parameter fitting every 5 rounds. The same strategy was used for the image classification experiments in the next section.

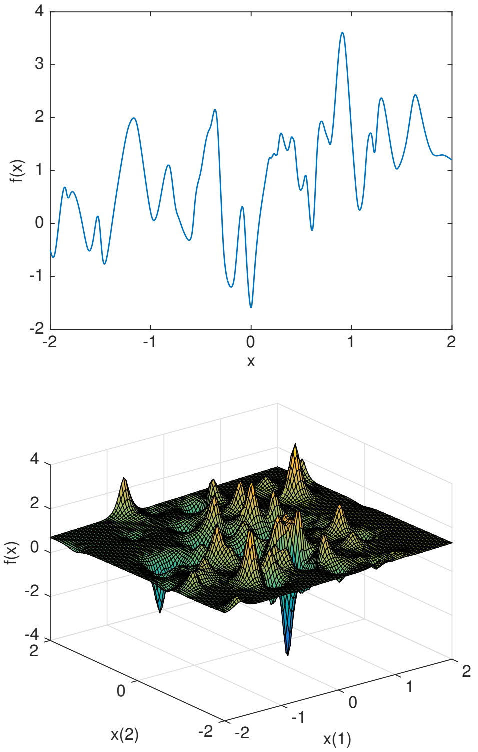

Figure 2: Examples of a function sampled from 1-D GP (top), and a function sampled from 2-D GP (bottom) with isotropic Matérn kernel and linear mean function. We deliberately create many local optimums to make the problem hard.

2.4 Parameter tuning for image classification

We use the linear SVM in the liblinear package for all the image classification experiments. We extract the FC7 activation from the imagenet reference network in the Caffe deep learning package [5] as the visual feature. The reported classification accuracy is the accuracy averaged over all the categories. '-c' cost is the model parameter we tune for the linear SVM.

In Caltech101 and Caltech256 experiment [1, 2], there are 8,677 images from 101 object categories in the Caltech101 and 29,780 images from 256 object categories. The training size is 30 images per category, and the rest are test images.

In SUN397 experiment [12], there are 108,754 images from 397 scene categories. Images are randomly split into training and test set. The training size is 50 images per category, and the rest are test images.

In MIT Indoor67 experiment [7], there are 15,620 images from 67 indoor scene categories. Images are randomly split into training set and test set. The training size is 100 images per category, and the rest are test images.

In Stanford Action40 experiment [13], there are 9,532 images from 40 action categories. Images are randomly split into training set and test set. The training size is 100 images per category, and the rest are test images.

In UIUC Event8 experiment [6], there are 1,579 images from 8 event categories. Images are randomly split into training set and test set. Training size is 70 images per category, and the rest are test images.

We used features extracted from a convolutional neural network (CNN) that was trained on images from ImageNet. It has been found [14] that features from a CNN trained on a set of images focused more on places than on objects, the Places database, work better in some domains. So, we repeated our experiments using the Places-CNN features, and the results are shown in Figure 3 and Table 1. All the methods help to improve the classification accuracy on the validation set. EST methods achieve good accuracy on each validation set on par with the best competitors for most of the datasets. And we also observe that for Caltech101 and Event8, Rand and UCB converge faster and achieve better accuracy than other methods. As we have shown in Section **??**, UCB and Rand perform worse than other methods in terms of cumulative regret, because they tend to explore too much. However, more exploration can be helpful for some black-box functions that do not satisfy our assumption that they are samples from GP. For example, for discontinuous step functions, pure exploration can be beneficial for simple regret. One possible explanation for the better results of Rand and UCB is that the black-box functions we optimize here are possibly functions not satisfying our assumption. The strong assumption on the black-box function is also a major drawback of Bayesian optimization,

2.5 Comparison to entropy search methods

Entropy search methods [3, 4] aim to minimize the entropy of the probability for the event $M_{\boldsymbol{x}}$ ($\boldsymbol{x} = \arg\max_{\boldsymbol{x}' \in \mathfrak{X}} f(\boldsymbol{x}')$). Although not suitable for minimizing cumulative regret, ES methods are intuitively ideal for minimizing simple regret. We hence in this section compare the empirical performance of entropy search (ES) [3] and predictive entropy search (PES) [4] to that of the EST methods (EST/GP-UCB/PI) and EI.

Since both ES and PES only support squared exponential covariance function and zero mean function in their code right now, and it requires significant changes in their code to accommodate other covariance functions, we created synthetic functions that are different from the ones we used in Section 4 in the paper. The new functions are sampled from 1-D (80 functions) and 2-D GP (20 functions) with squared exponential kernel ($\sigma_f = 0.1$ and $l = 1$) and 0 mean. Function examples are shown in Figure 4.

We show the results on these synthetic functions in Figure 5,6, and a standard optimization test function, Branin Hoo function, in Figure 7. It is worth noting that ES methods make queries on the most informative points, which are not necessarily the points with low regret. At each round, ES methods make a "query" on the black-box function, and then make a "guess" of the $\arg\max$ of the function (but do not test the "guess"). We plot the regret achieved by the "guesses" made by ES methods. For the 1-D GP task, all the methods behave similarly and achieve zero regret except RAND. For the 2-D GP task, EI is the fastest method to converge to zero regret, and in the end ESTn, PI,EI and ES methods achieve similar results. For the test on Branin Hoo function, PES achieves the lowest regret. ESTa converges slightly faster than PES, but to a slightly higher regret.

We also compared the running time for all the methods in Table 2. [1] It is assumed in GP optimization that it is more expensive to evaluate the blackbox function than computing the next query to evaluate using GP optimization techniques. However, in practice, we still want the algorithm to output the next query point as soon as possible. For ES methods, it can be sometimes unacceptable to run them for black-box functions that take minutes to complete a query.

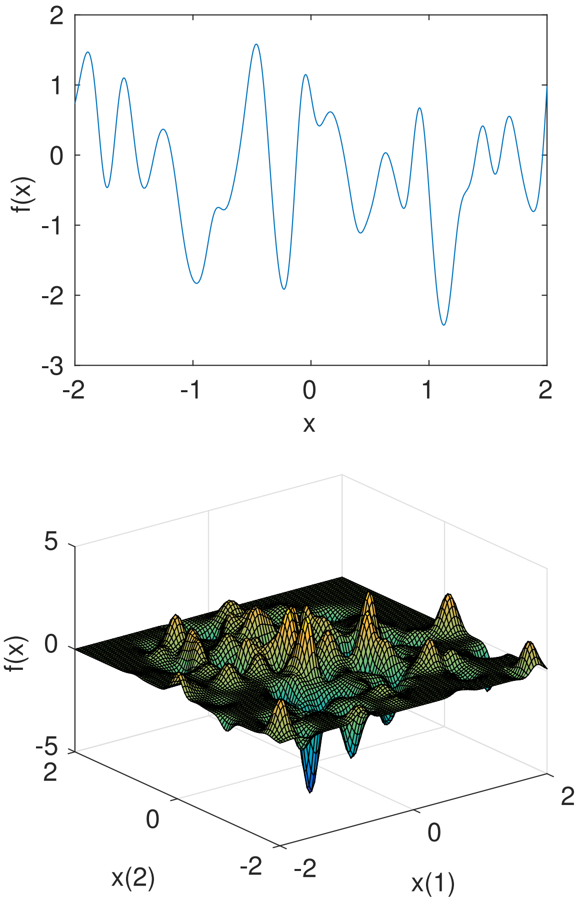

Figure 4: Examples of a function sampled from 1-D GP (left), and a function sampled from 2-D GP (right) with squared exponential kernel and 0 mean functions. These functions can be easier than the ones in Figure 2 since they have fewer local optima.

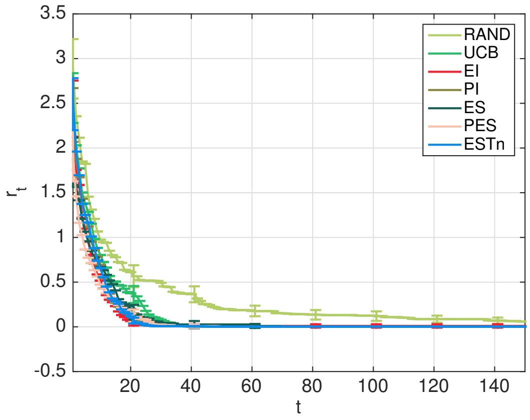

Figure 5: Simple regret for functions sampled from 1-D GP with squared exponential kernel and 0 mean.

References

[1] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 ob-

[1] All of the methods were run with MATLAB (R2012b), on Intel(R) Xeon(R) CPU E5645 @ 2.40GHz.

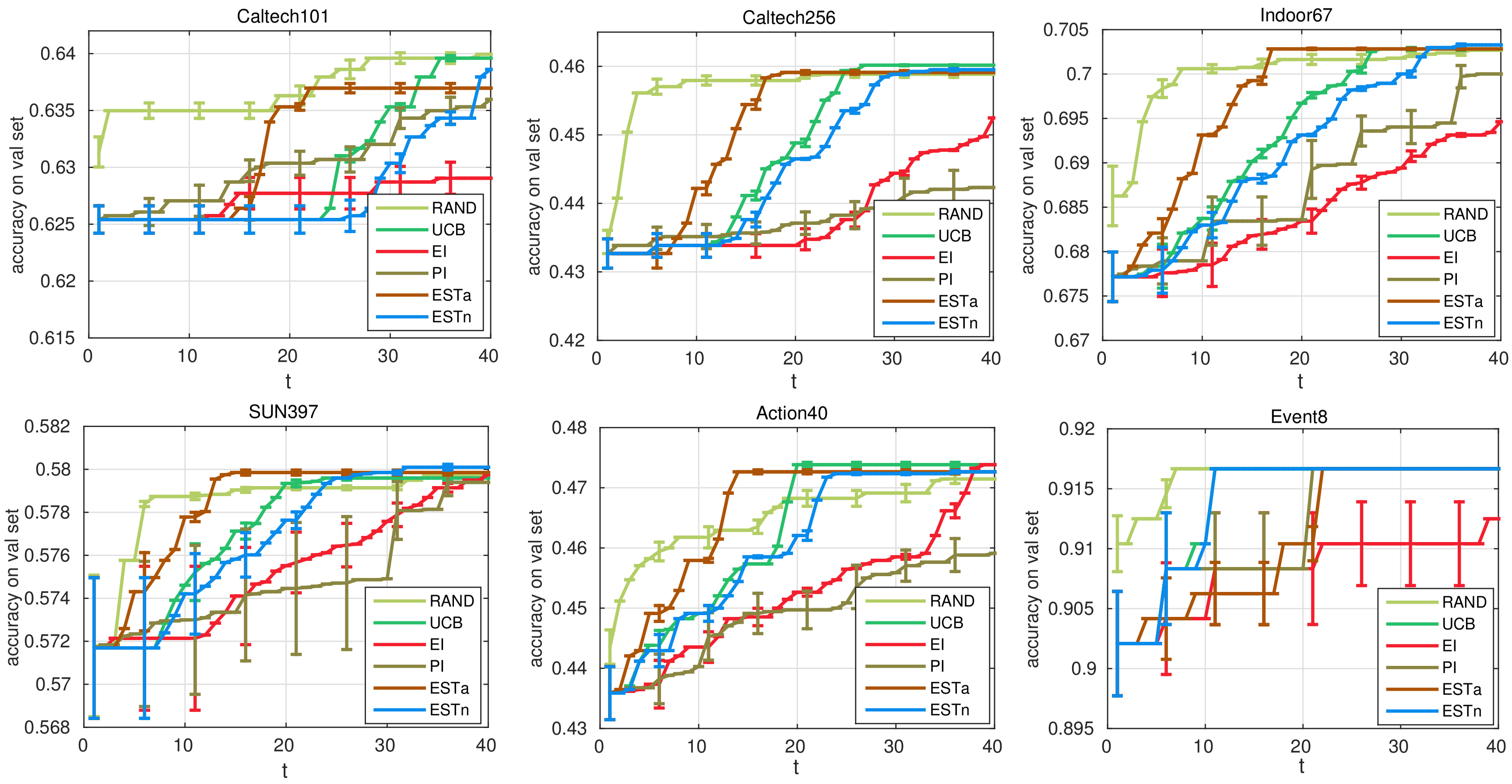

Figure 3: Maximum accuracy on the validation set over iteration of the optimization. Experiments are repeated 5 times. The visual features used here are Deep CNN features pre-trained on the Places database.

Table 1: Classification accuracy on the test set of the datasets after the model parameter is tuned by ESTa and ESTn. Tuning achieves good improvement over the results in [14].

	Caltech101	Caltech256	Indoor67	SUN397	Action40	Event8
Imagenet-CNN feature	87.22	67.23	56.79	42.61	54.92	94.42
ESTa	88.23	69.39	60.02	47.13	57.60	94.91
ESTn	88.25	69.39	60.08	47.21	57.58	94.86
Places-CNN feature	65.18	45.59	68.24	54.32	42.86	94.12
ESTa	66.94	47.51	70.27	58.57	46.24	93.79
ESTn	66.95	47.43	70.27	58.65	46.17	93.56

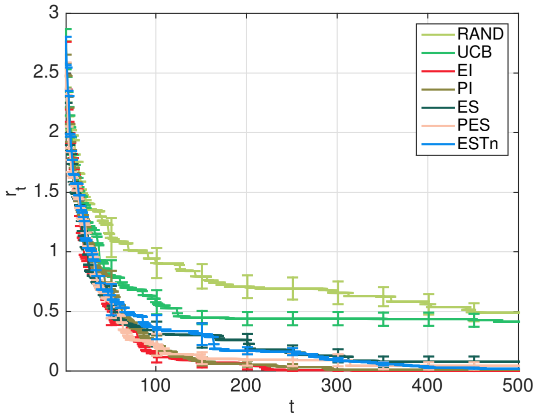

Figure 6: Simple regret for functions sampled from 2-D GP with squared exponential kernel and 0 mean.

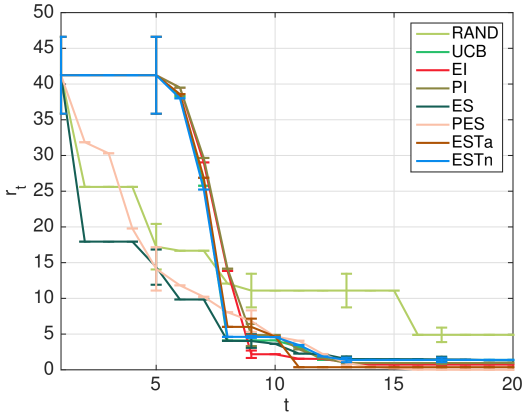

Figure 7: Simple regret for Branin Hoo function. UCB/EI/PI/ESTa/ESTn use the isotropic Matérn kernel with parameters $\ell = 0.1, \sigma_f = 1$; ES/PES use the isotropic squared exponential kernel with $\ell = 0.1, \sigma_f = 1$.

Table 2: Comparison on the running time (s) per iteration.

RAND	UCB	EI	PI
0.0002	0.075	0.079	0.076
ESTa	ESTn	ES	PES
0.078	0.55	56	106

ject categories. *Computer Vision and Image Understanding*, 2007.

[2] Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. 2007.

[3] P. Hennig and C. Schuler. Entropy search for information-efficient global optimization. *JMLR*, 13:1809–1837, 2012.

[4] José Miguel Hernández-Lobato, Matthew W Hoffman, and Zoubin Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. In *Advances in Neural Information Processing Systems*, 2014.

[5] Yangqing Jia. Caffe: An open source convolutional architecture for fast feature embedding. `http://caffe.berkeleyvision.org/`, 2013.

[6] Li-Jia Li and Li Fei-Fei. What, where and who? classifying events by scene and object recognition. In *Proc. ICCV*, 2007.

[7] A Quattoni and A Torralba. Recognizing indoor scenes. In *Proc. CVPR*, 2009.

[8] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, pages 2951–2959, 2012.

[9] Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *ICML*, 2010.

[10] Russ Tedrake. *Underactuated Robotics: Algorithms for Walking, Running, Swimming, Flying, and Manipulation (Course Notes for MIT 6.832)*. Downloaded in Fall, 2014 from `http://people.csail.mit.edu/russt/underactuated/`.

[11] Russ Tedrake. Drake: A planning, control, and analysis toolbox for nonlinear dynamical systems, 2014.

[12] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *Proc. CVPR*, 2010.

[13] Bangpeng Yao, Xiaoye Jiang, Aditya Khosla, Andy Lai Lin, Leonidas Guibas, and Li Fei-Fei. Human action recognition by learning bases of action attributes and parts. In *Proc. ICCV*, 2011.

[14] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *Advances in Neural Information Processing Systems*, pages 487–495, 2014.