University of Glasgow

Bayesian optimisation

Learning on a budget

Benefits of Gaussian processes:

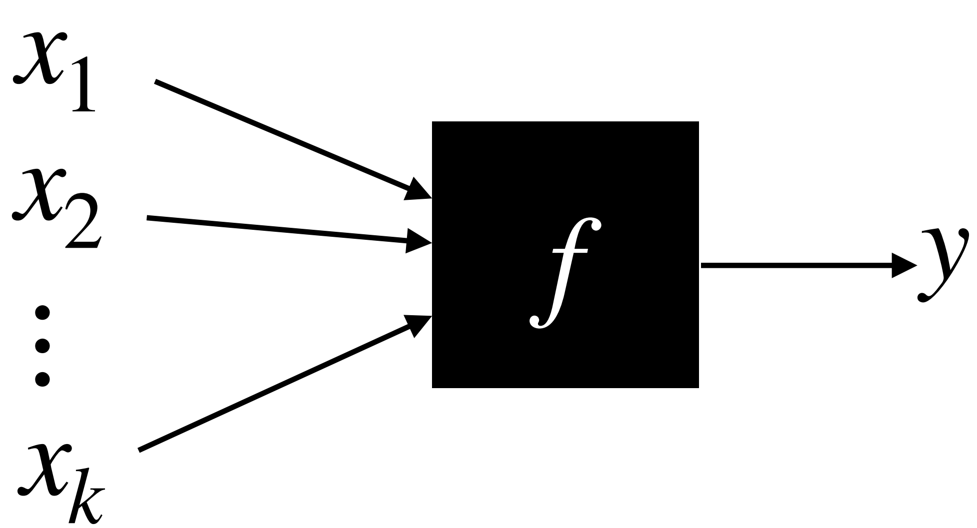
- Data efficient
- Robust uncertainty quantification
- Principled model evaluation (marginal likelihood)

Use case:

Optimising a function with only a handful of queries

Use case:

Optimising a function with only a handful of queries



Use case:

Optimising a function with only a handful of queries

Examples:

- Prosthetics design [1]
- Preference learning in animation [2]
- Hearing-aid personalisation [3]
- Neural network design [4]

[1] Kim et al. "Human-in-the-loop Bayesian optimization of wearable device parameters.” 2017.
[2] Brochu et al. "A Bayesian interactive optimization approach to procedural animation design.” 2010.
[3] Nielsen et al. "Hearing aid personalization." 2013.
[4] Snoek et al. "Practical bayesian optimization of machine learning algorithms." 2012.

Problem characteristics:

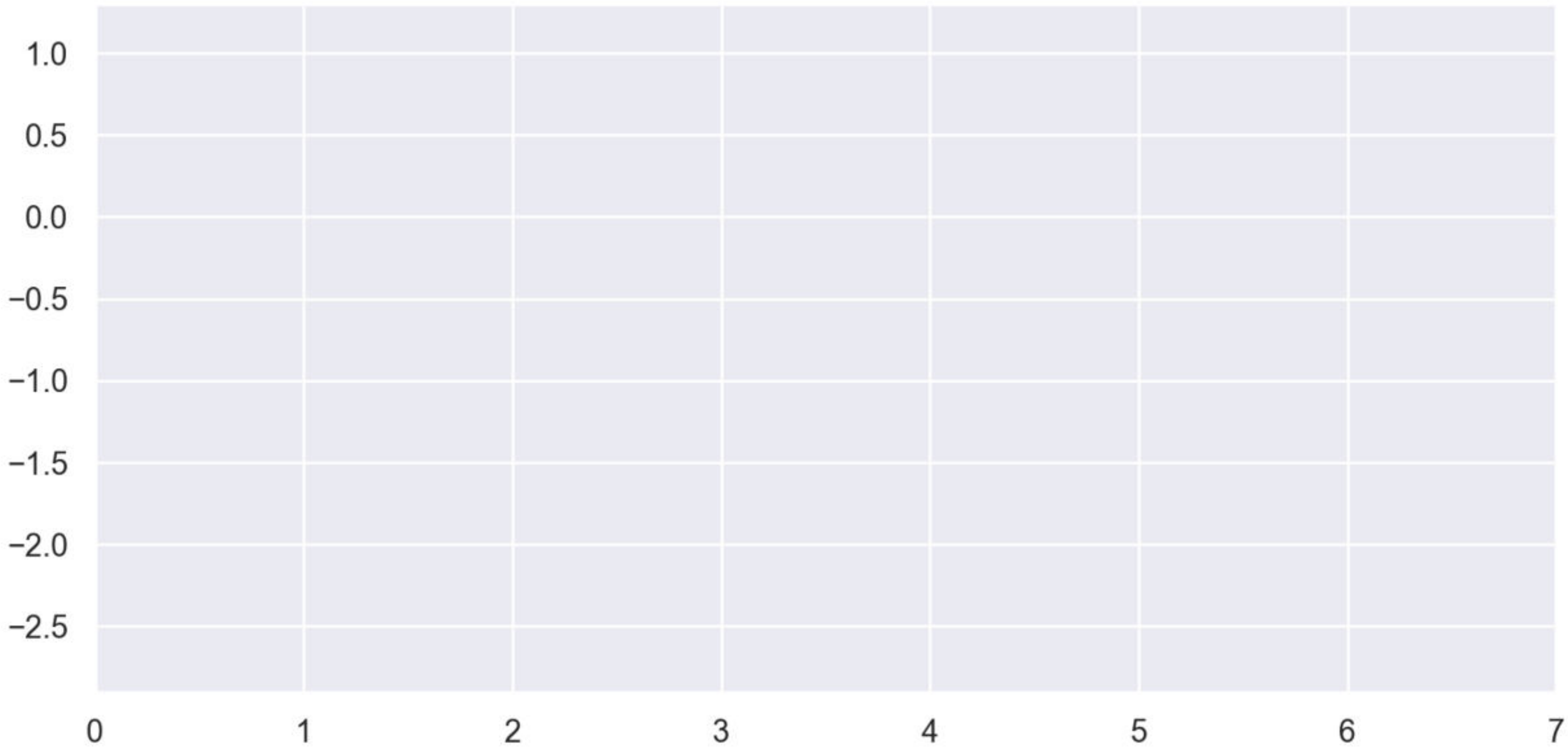
1. Evaluations are expensive (time, money, invasiveness)
2. The function is unknown
3. Evaluations *may* be noisy

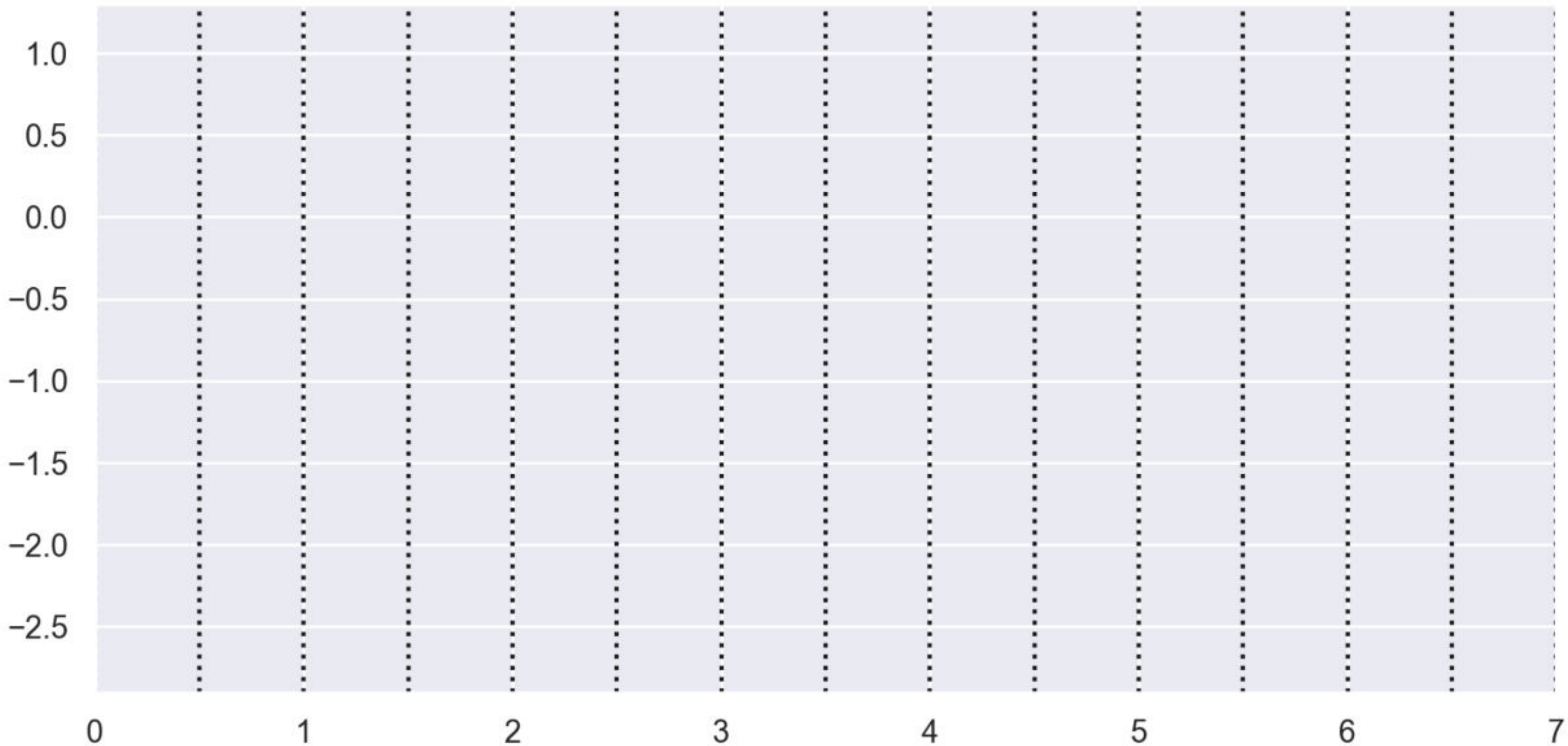
Requirement: Output must be smooth w.r.t. the inputs

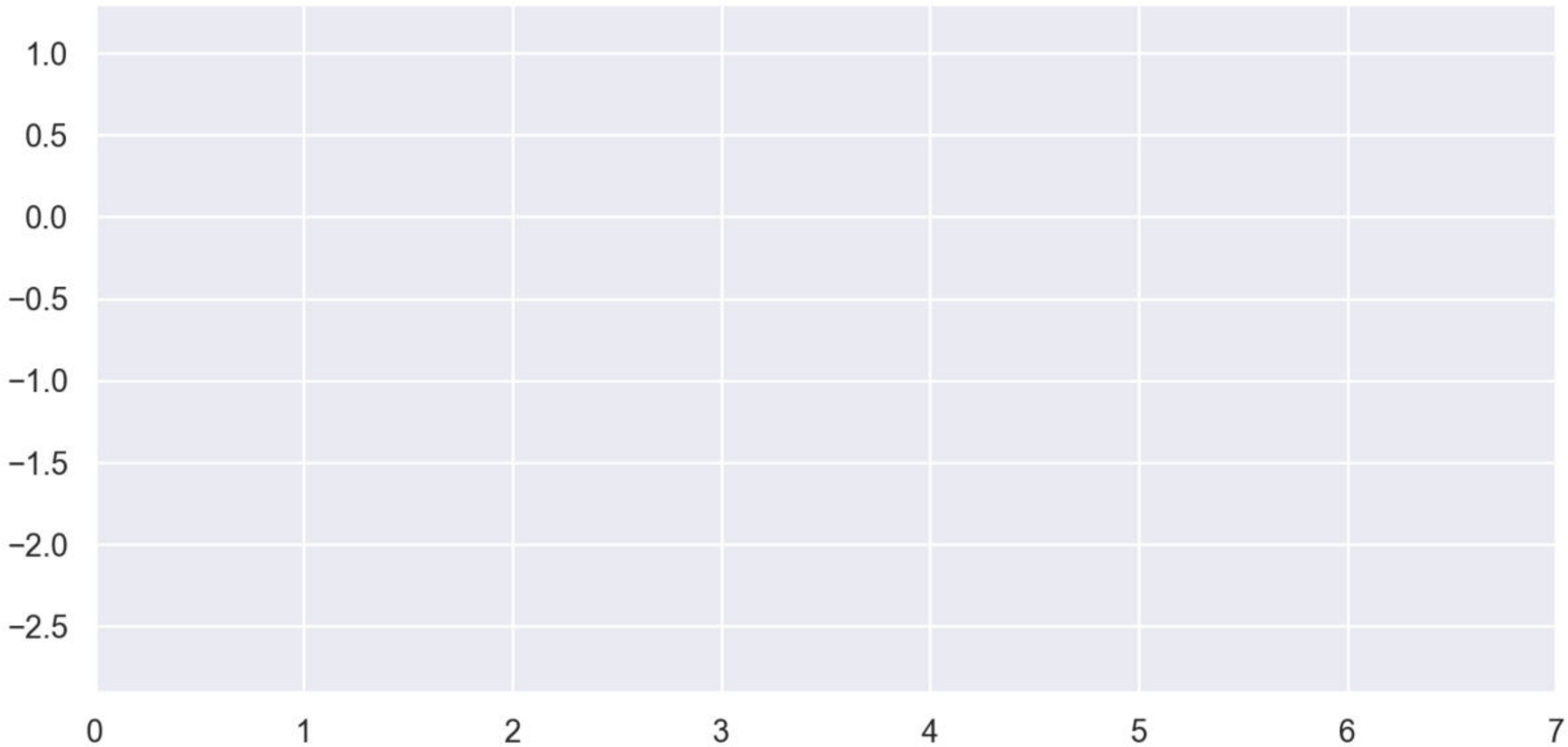
Neural network design:

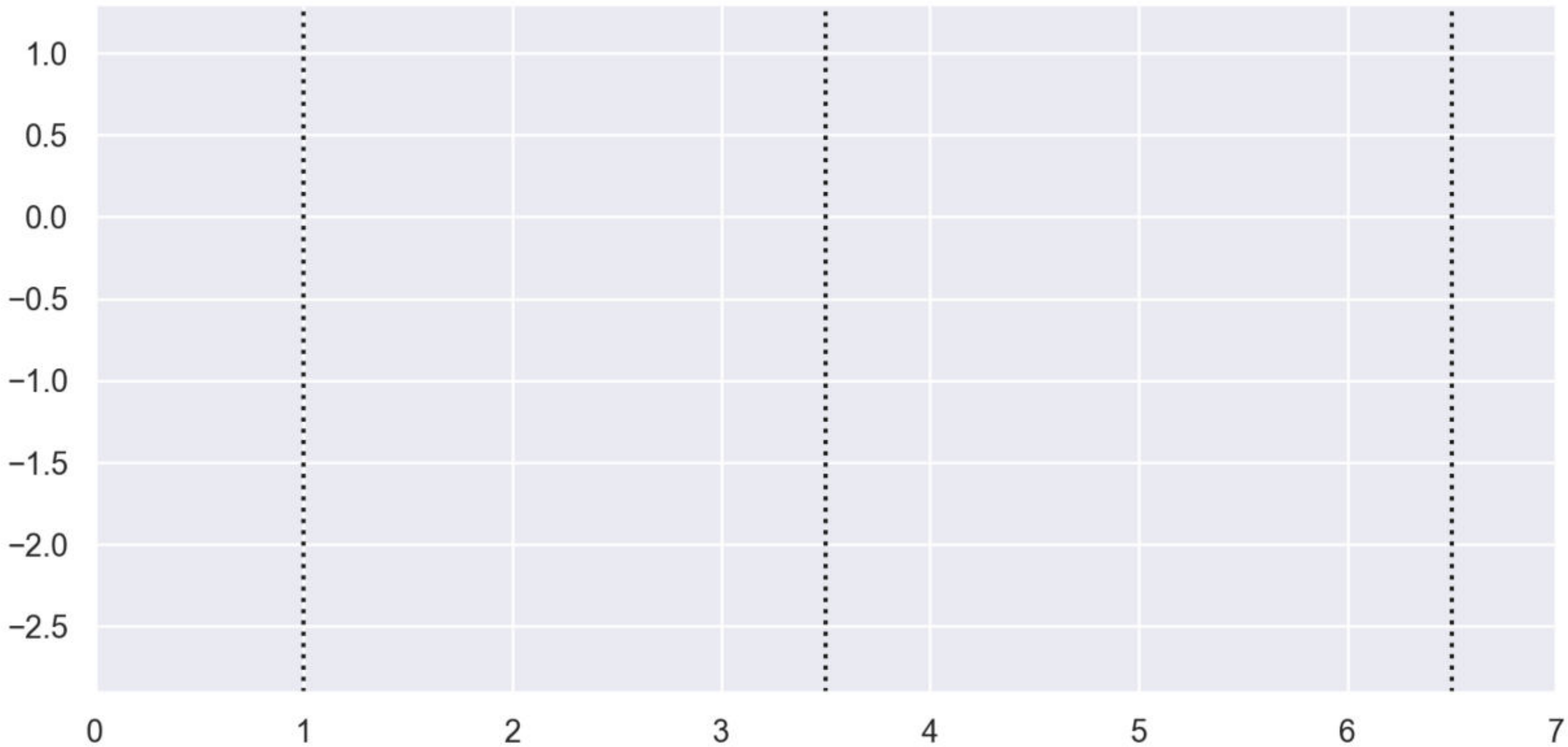
- Number of layers
- Number of units in each layer
- L1/L2 regularisation of weights or gradients
- Dropout
- Learning rate
- Momentum

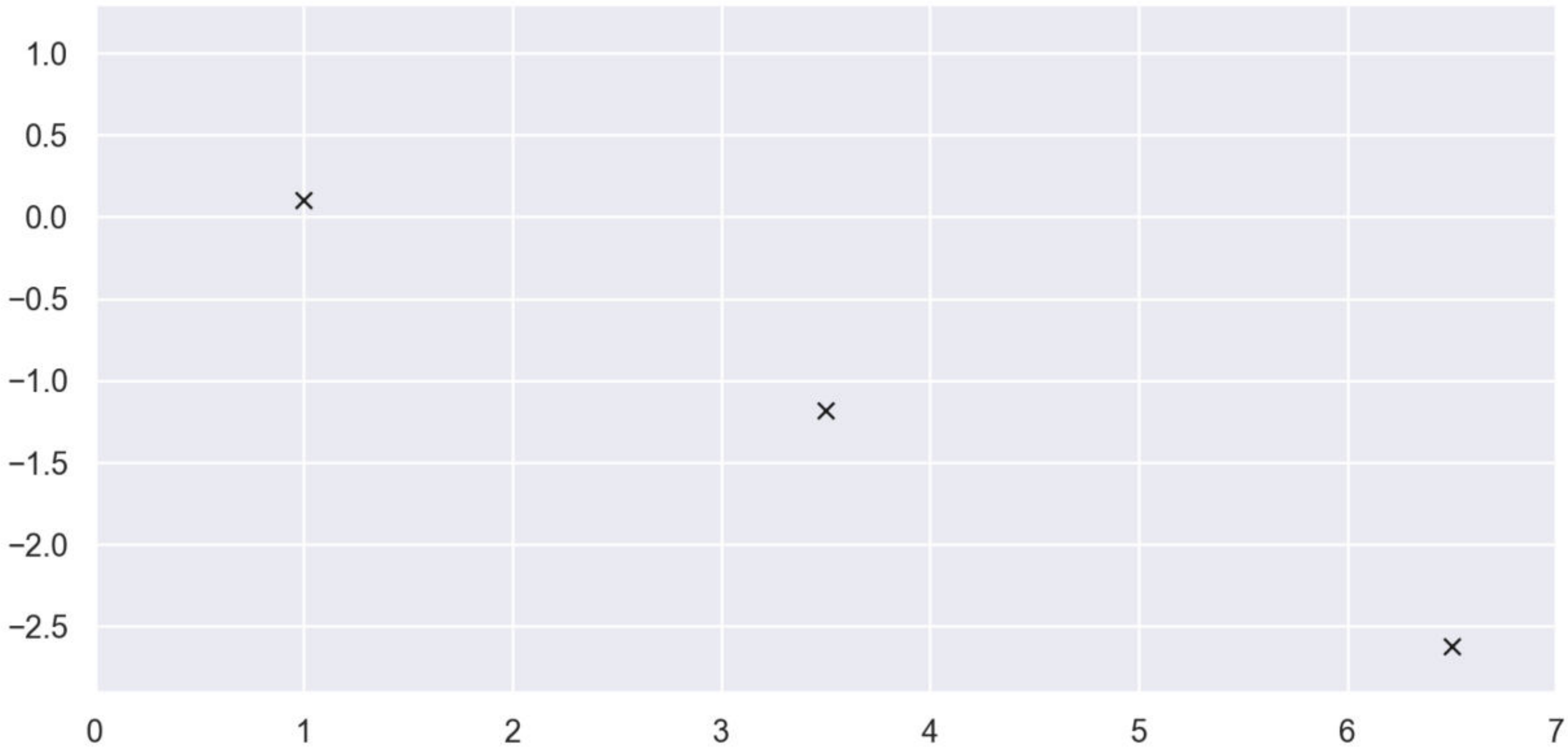
Chen, Yutian, et al. "Bayesian optimization in AlphaGo", 2018.


1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0
1
2
3
4
5
6
7


1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0
1
2
3
4
5
6
7


1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0
1
2
3
4
5
6
7


1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0
1
2
3
4
5
6
7

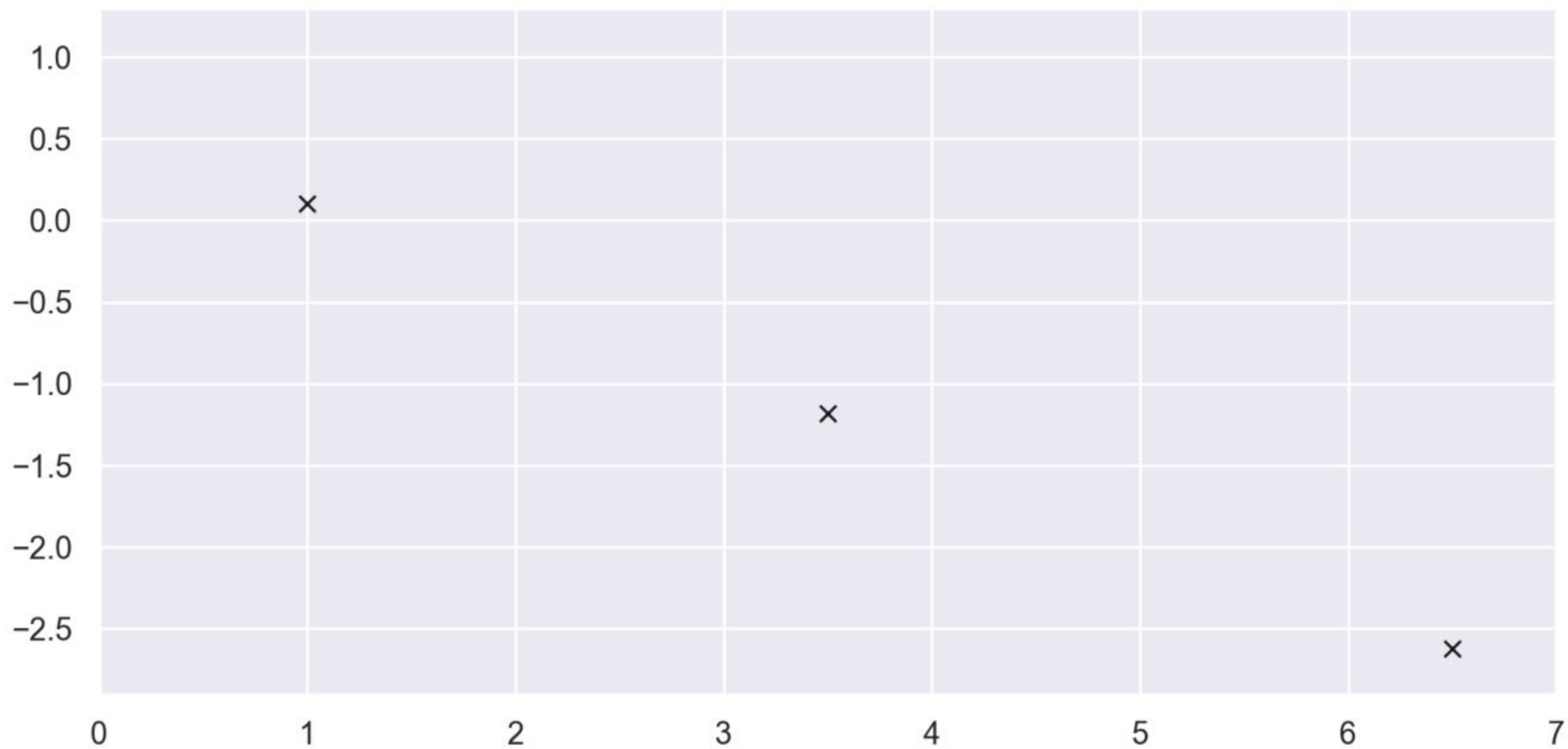

1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0
1
2
3
4
5
6
7

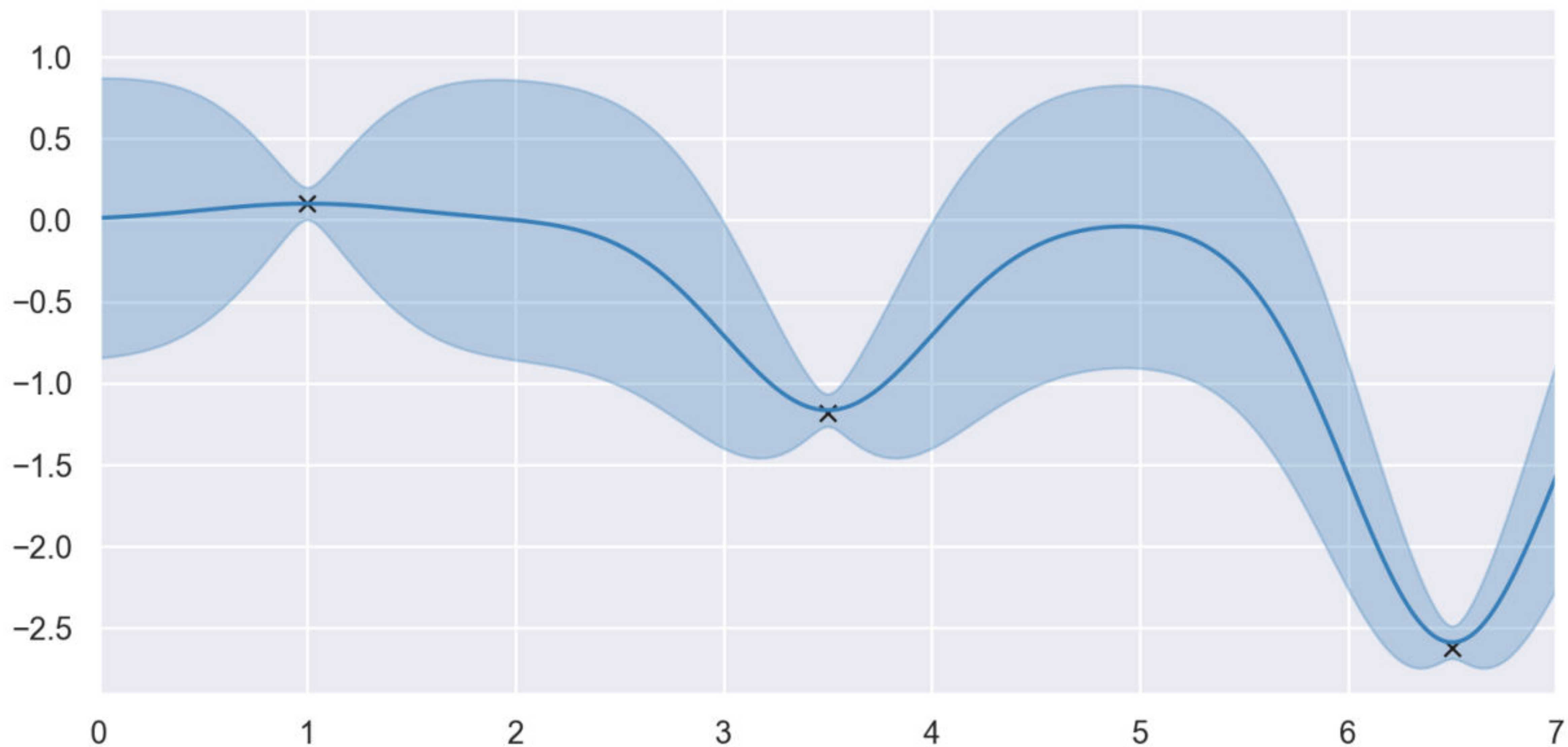

1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0
1
2
3
4
5
6
7

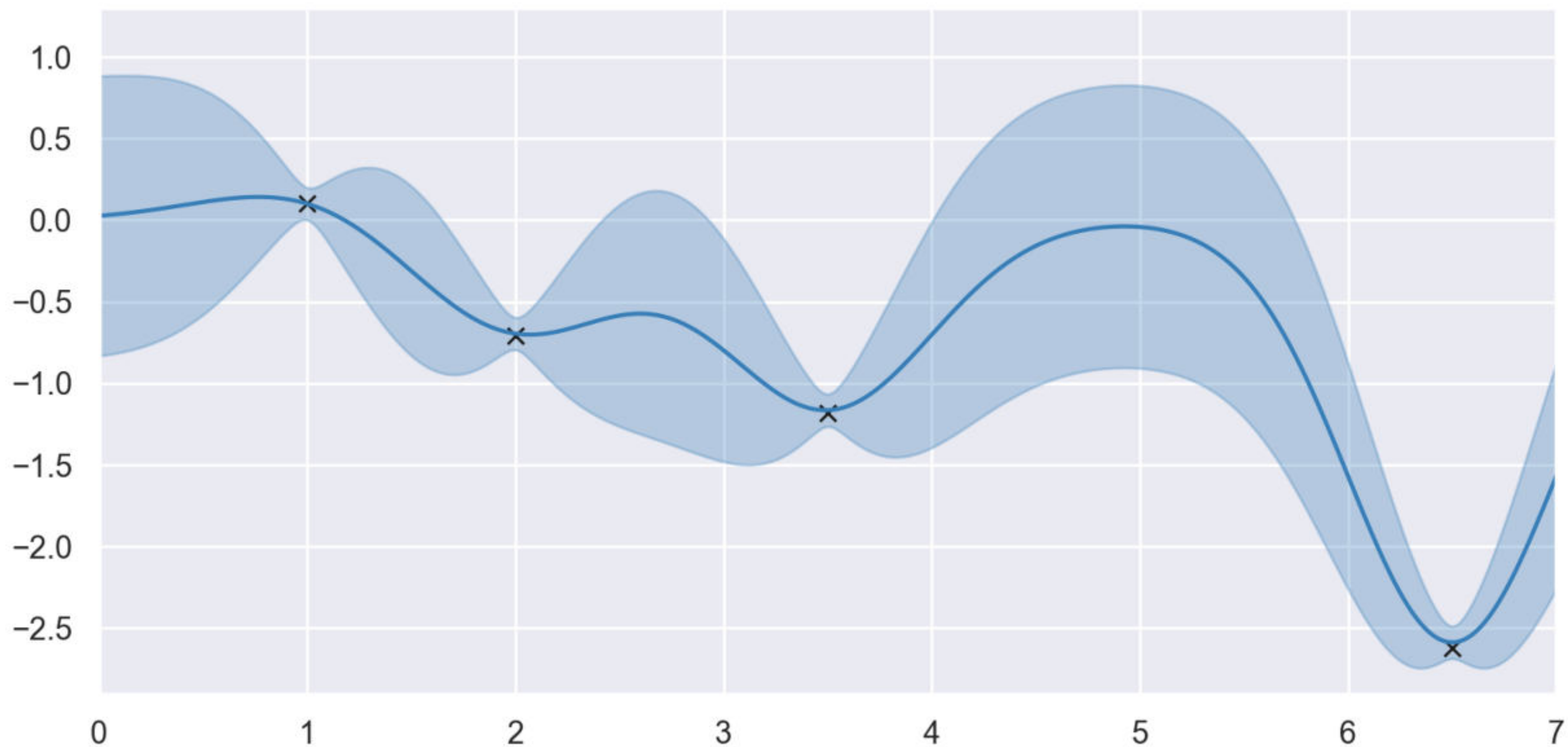

1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0
1
2
3
4
5
6
7

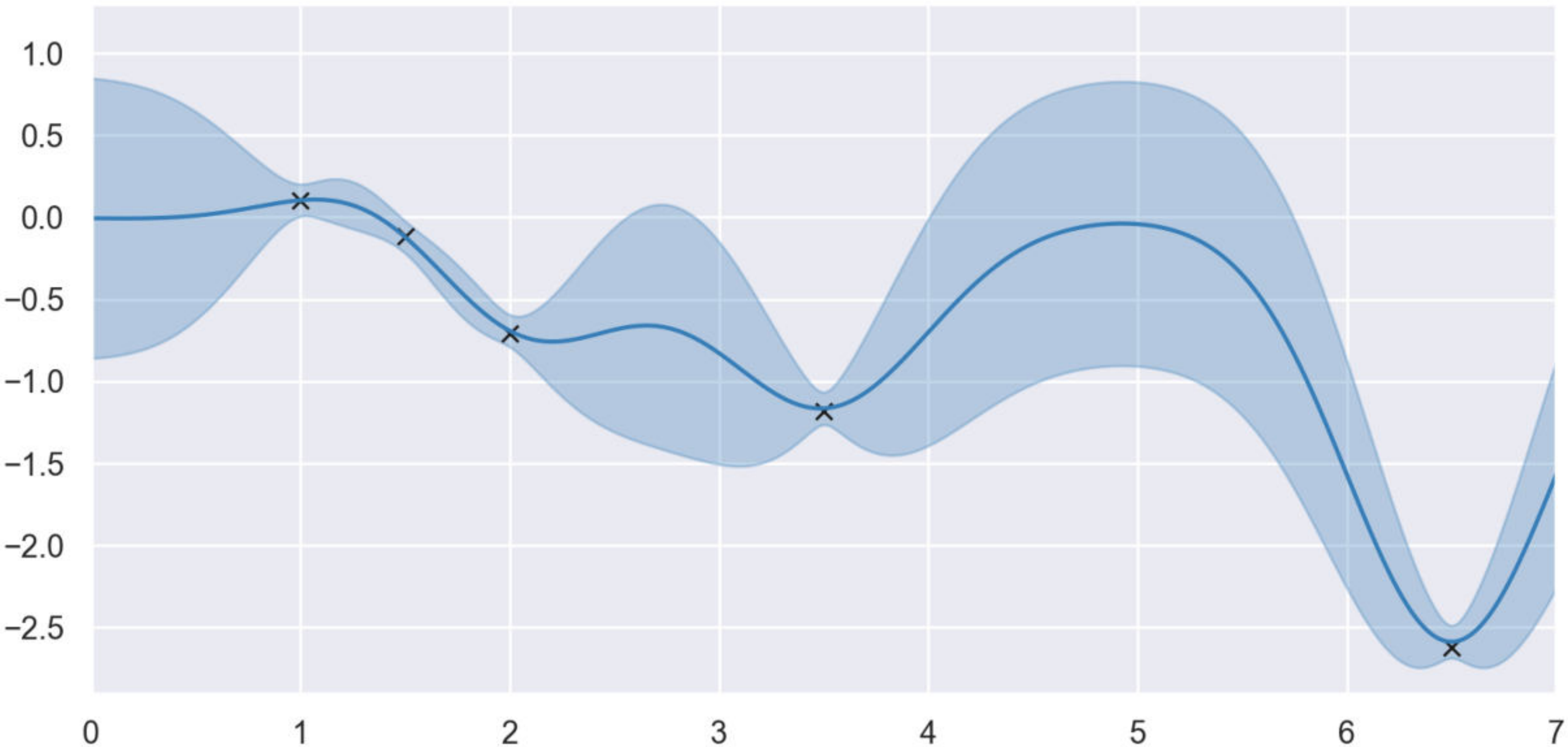
Optimisation concepts:

- Informed guess about un-evaluated inputs (prediction)
- Uncertainty
- Exploration/exploitation trade-off


1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0
1
2
3
4
5
6
7


1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0
1
2
3
4
5
6
7


1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0
1
2
3
4
5
6
7



1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
0
1
2
3
4
5
6
7

Optimisation concepts:

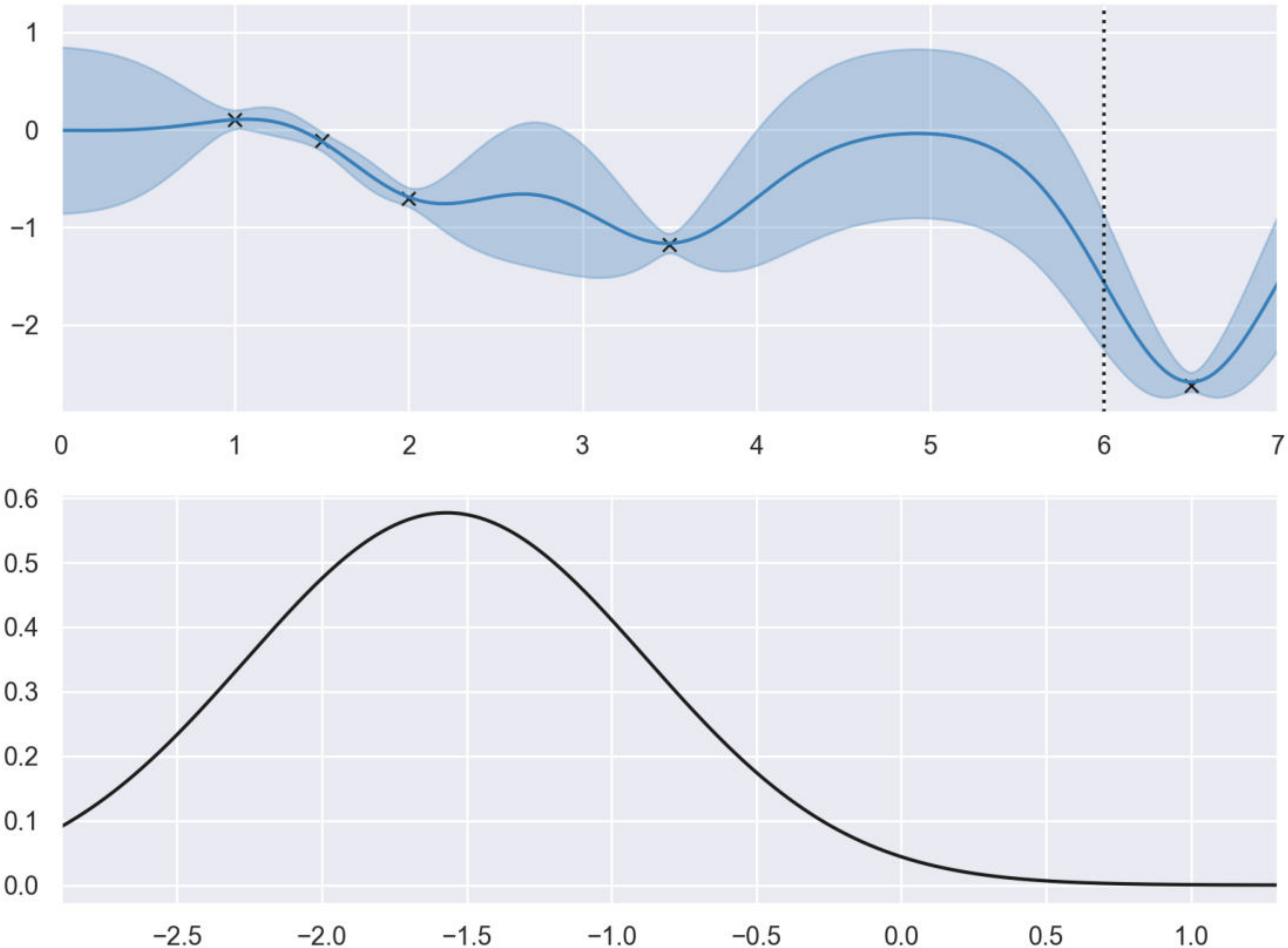
- Informed guess about un-evaluated inputs (prediction)
- Uncertainty
- Exploration/exploitation trade-off

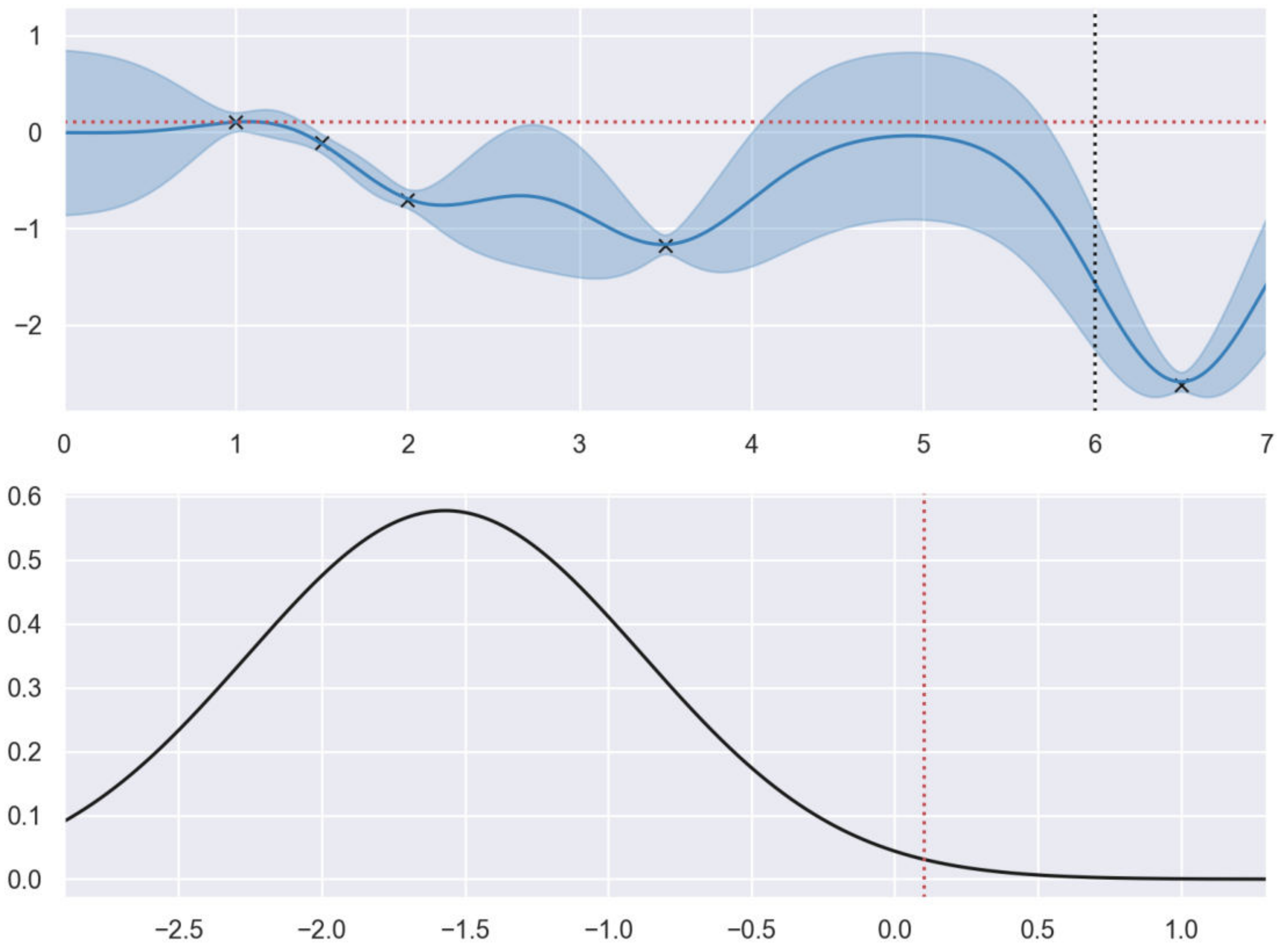
Acquisition function:

Yields the utility of a query, reliant on a predictive distribution.

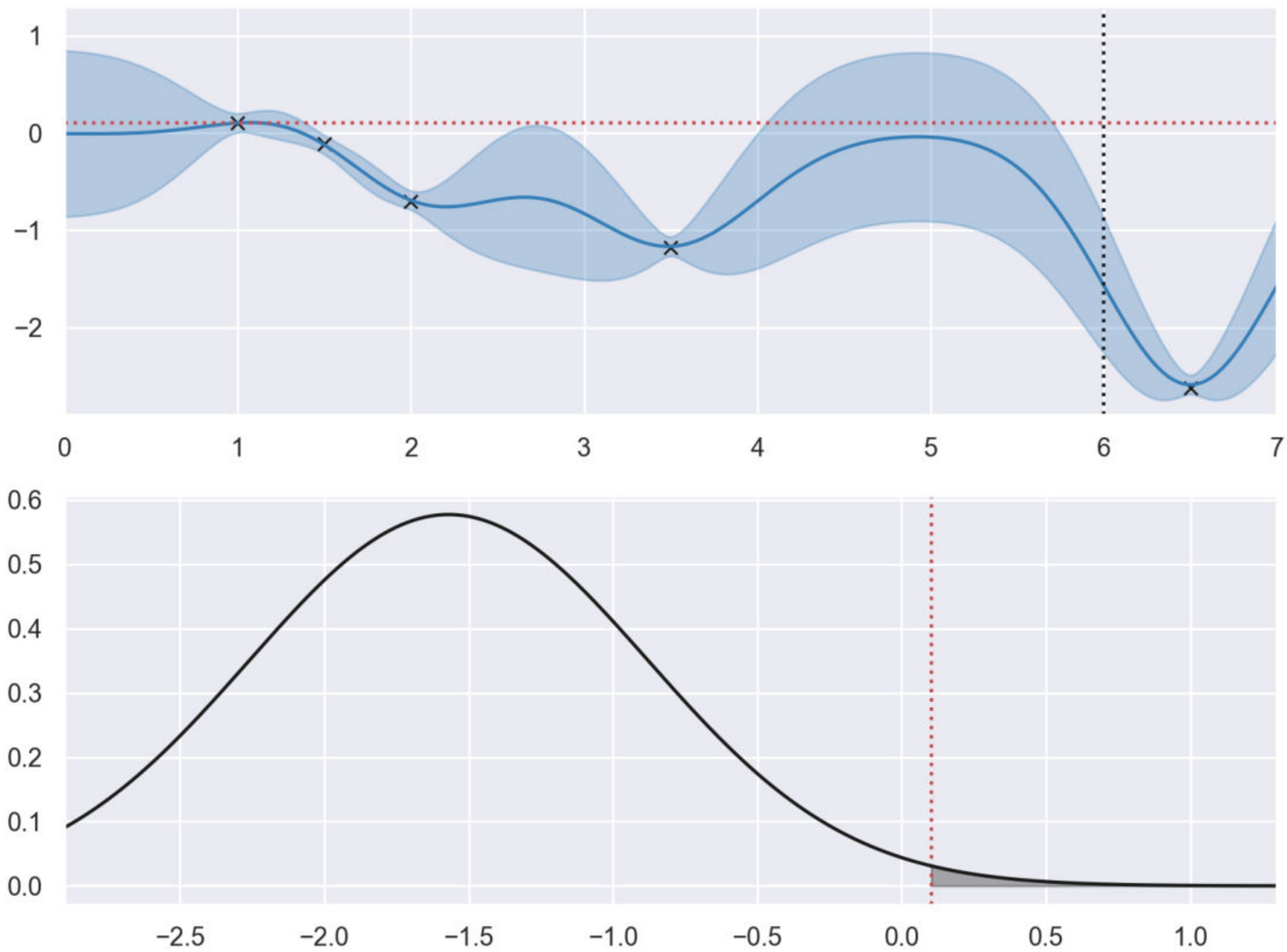
Expected Improvement:

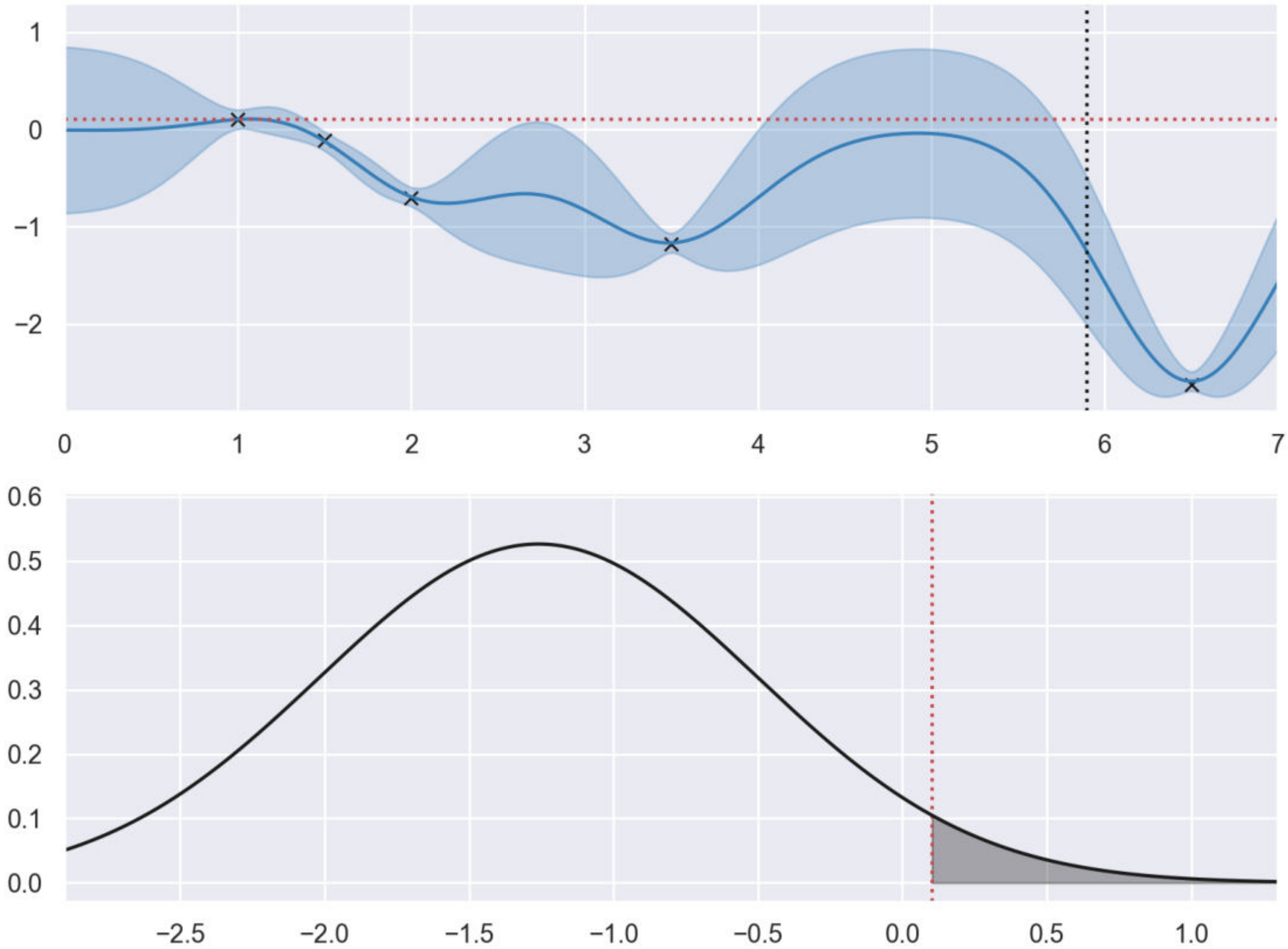
$$\alpha_{EI}(x; f^\star) = \int_{f^\star}^{\infty} f \cdot p(f \mid x, \mathcal{D})\, df$$


1
0
−1
−2
0
1
2
3
4
5
6
7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0

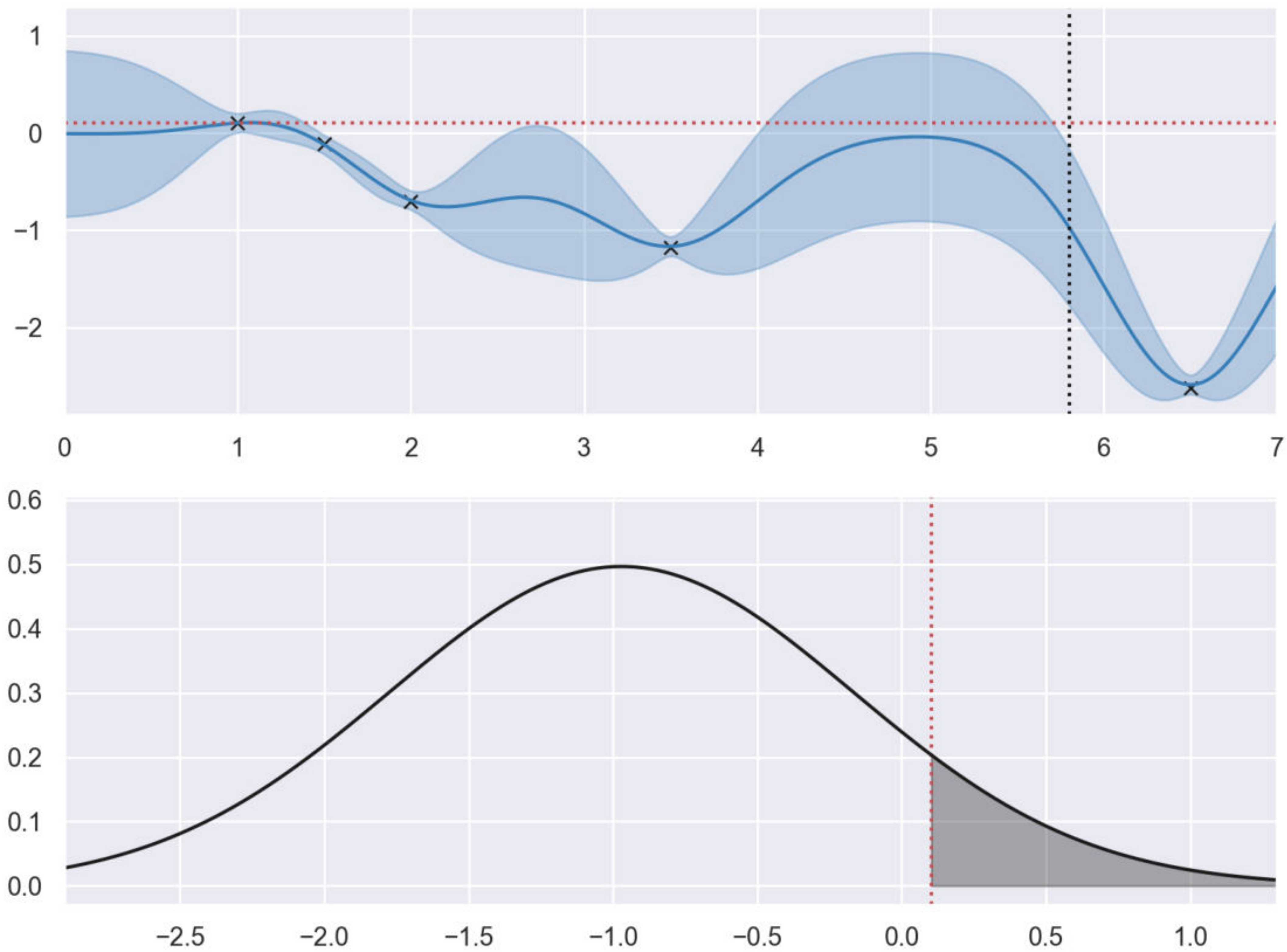


1
0
−1
−2
0
1
2
3
4
5
6
7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0

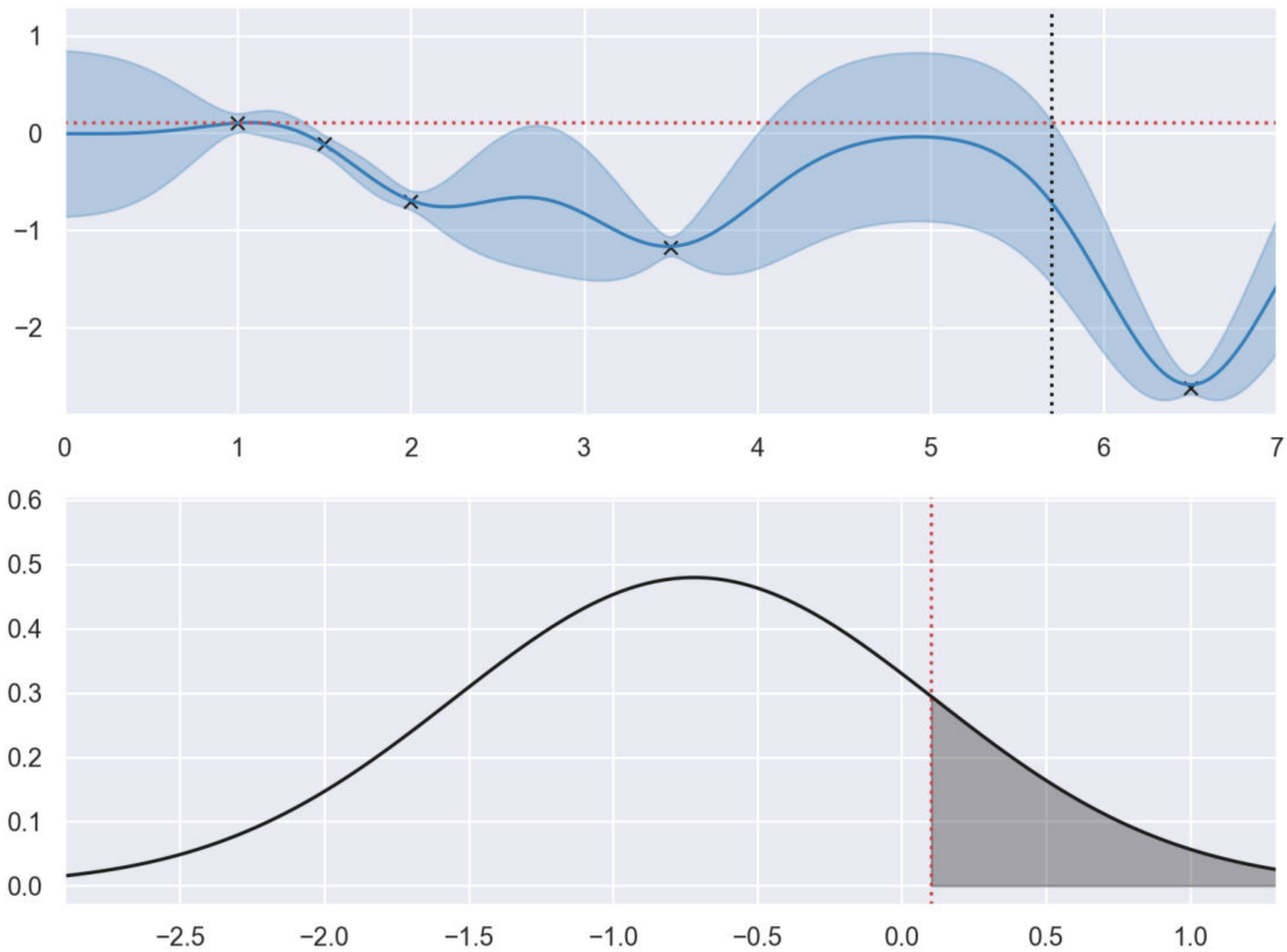

1
0
−1
−2
0
1
2
3
4
5
6
7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0



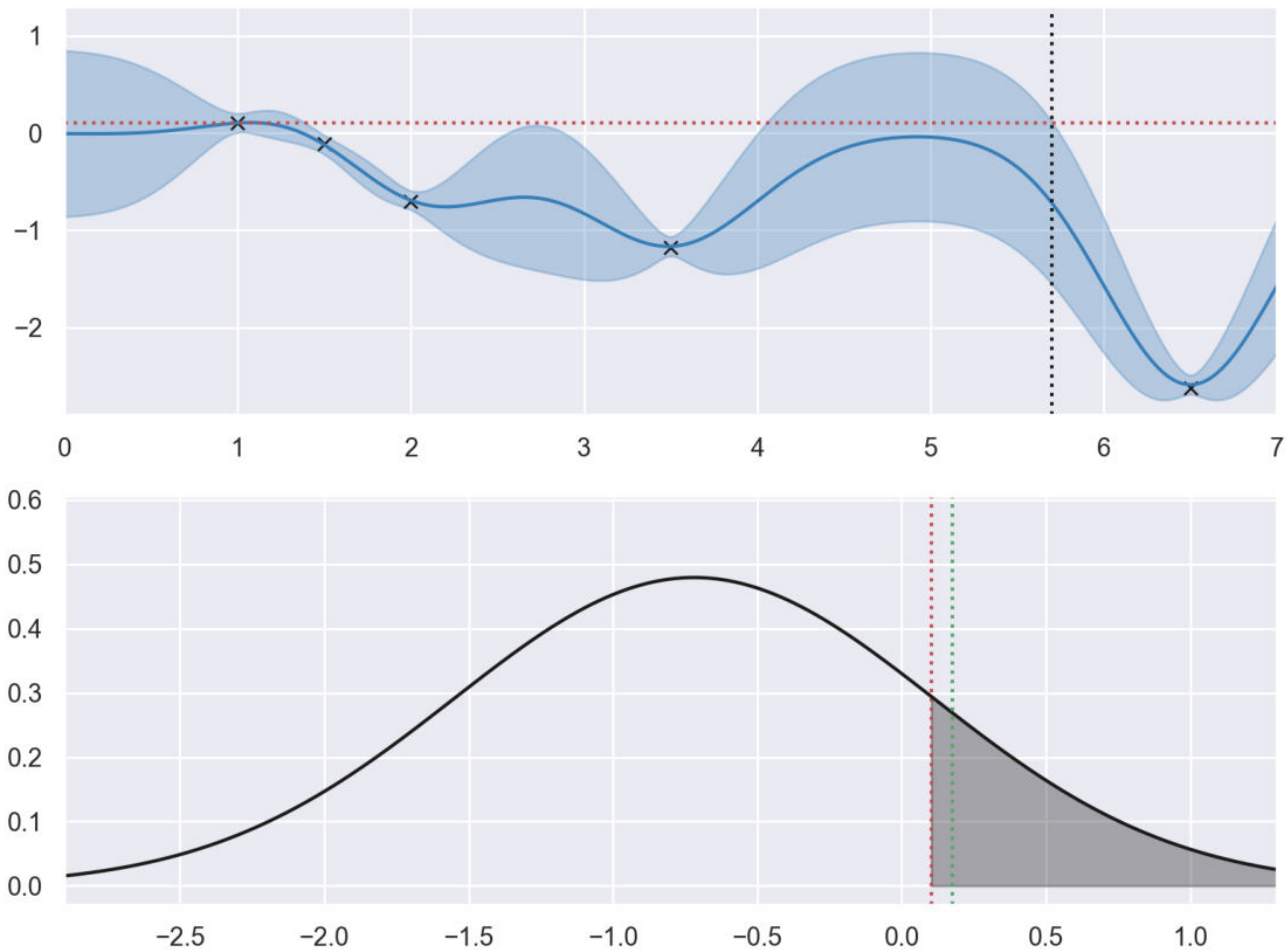
1
0
−1
−2
0
1
2
3
4
5
6
7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0



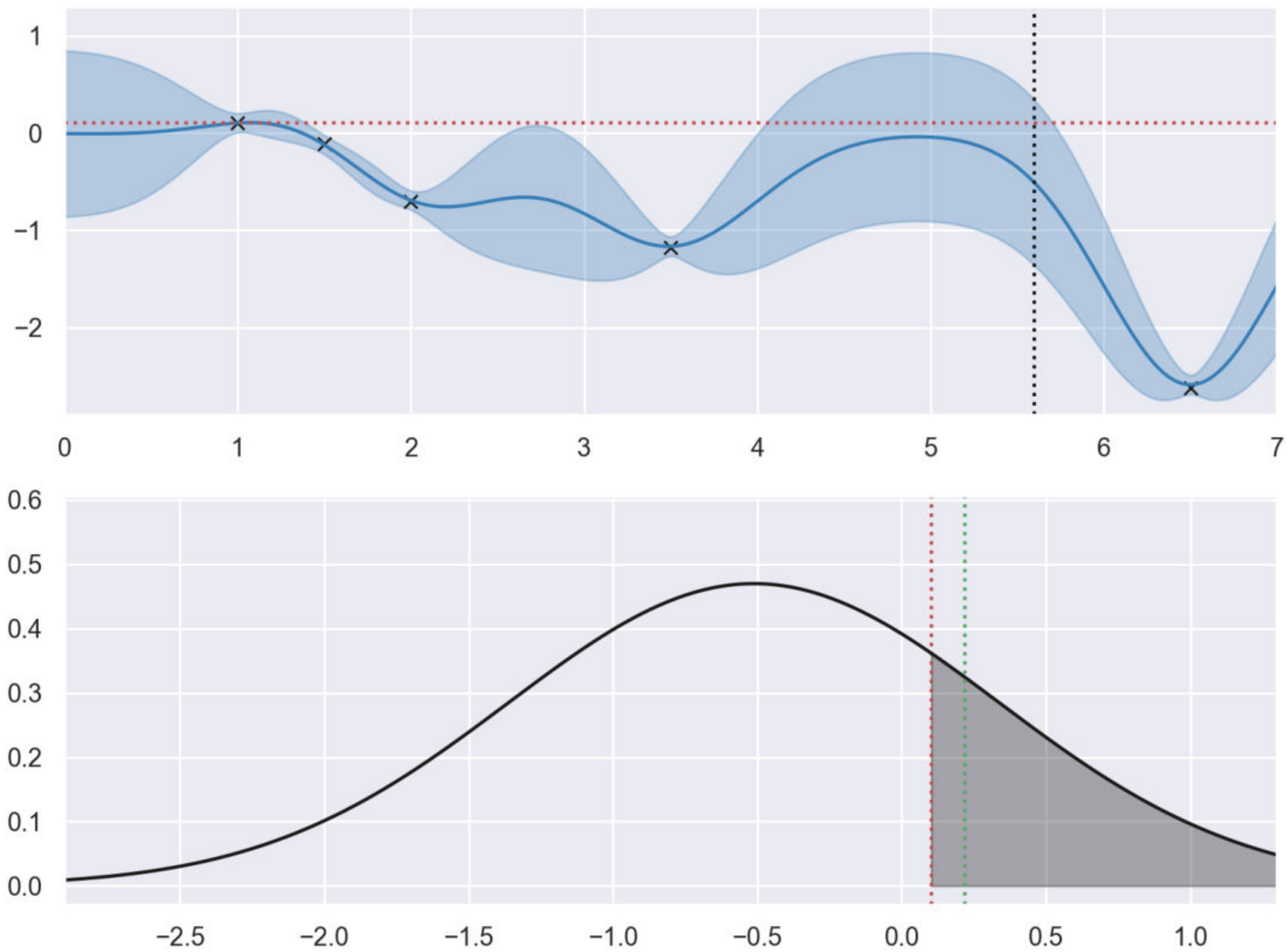
1
0
−1
−2
0
1
2
3
4
5
6
7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0



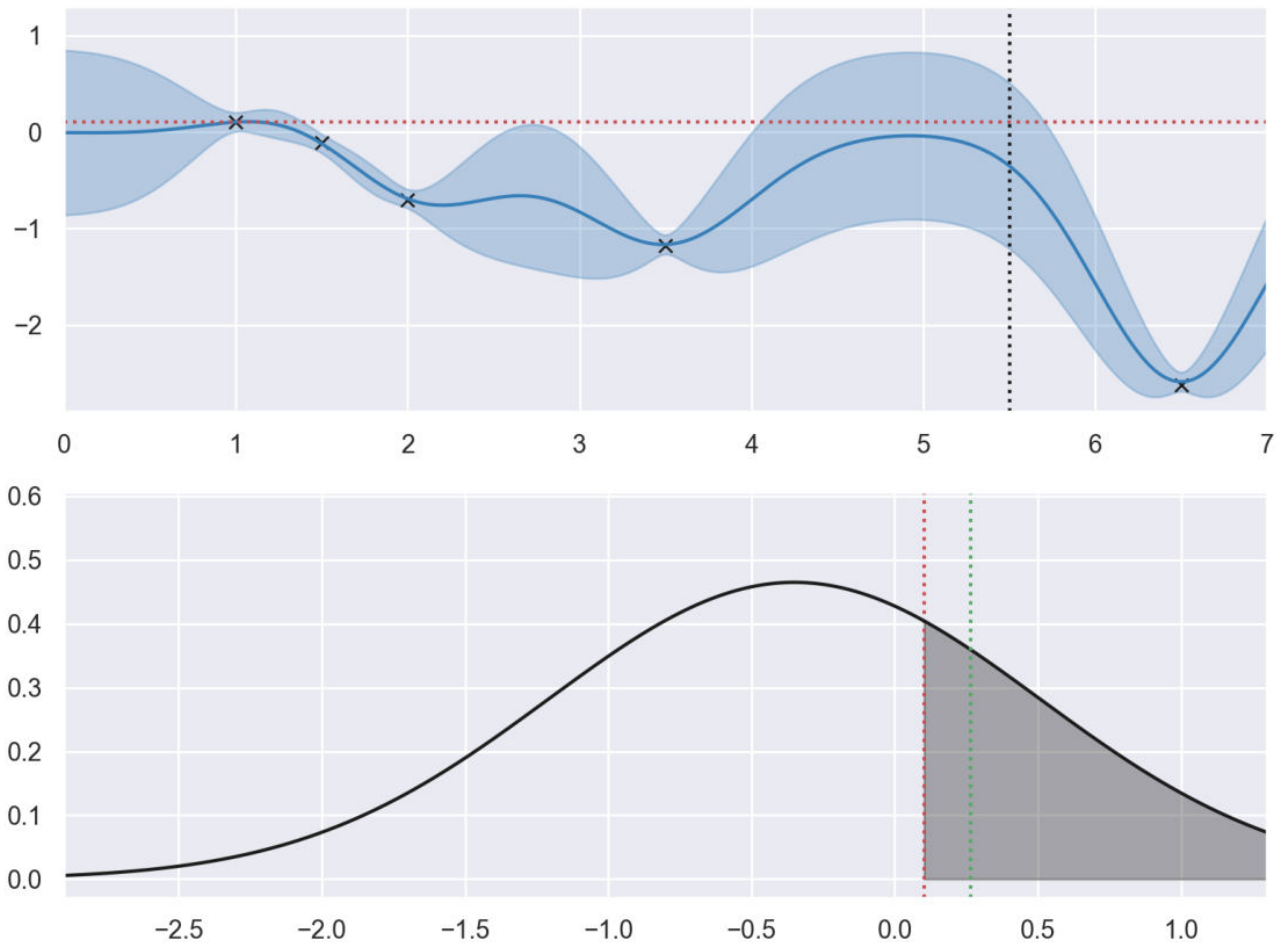
1
0
−1
−2
0
1
2
3
4
5
6
7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0



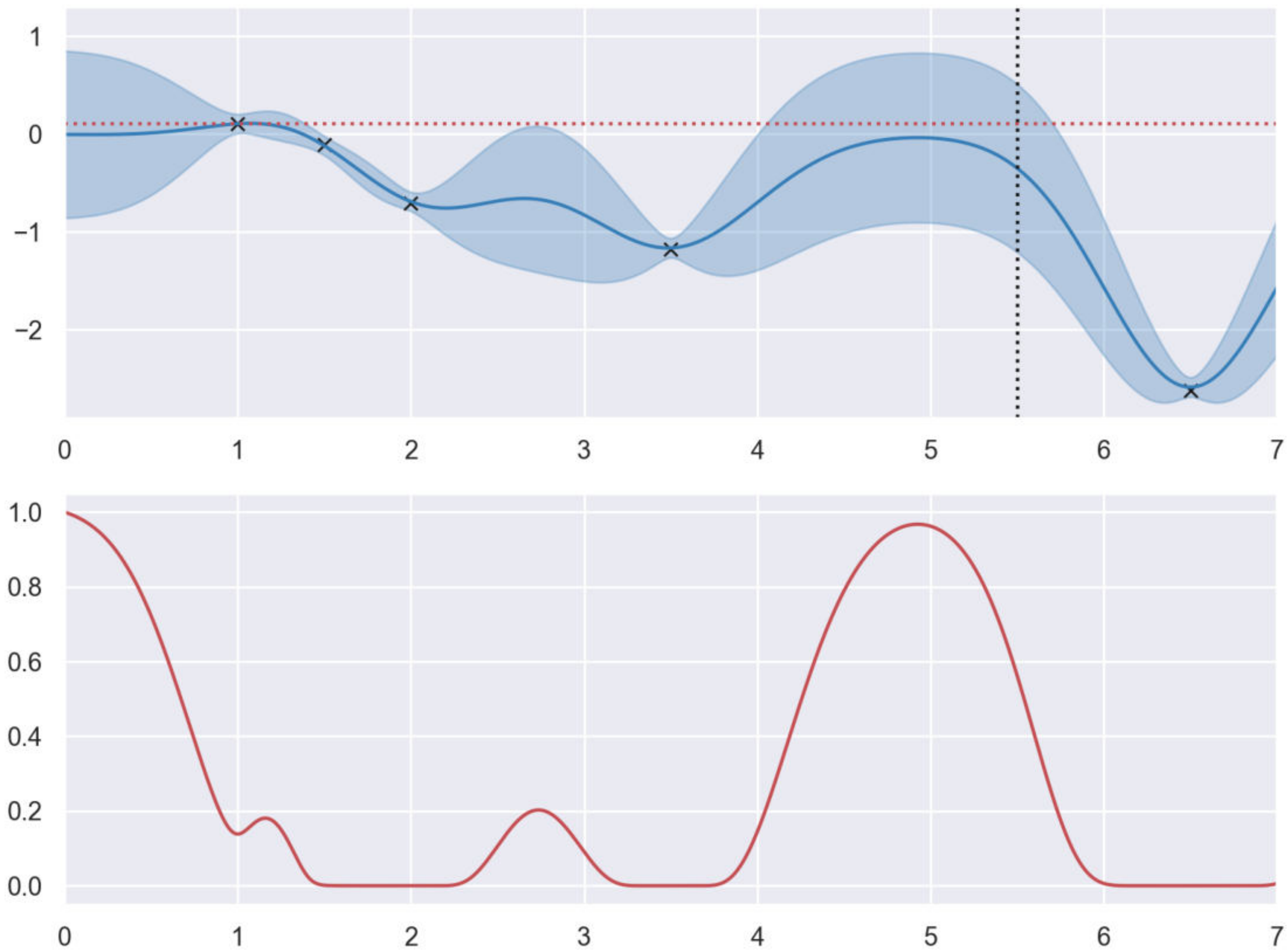
1
0
−1
−2
0
1
2
3
4
5
6
7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0

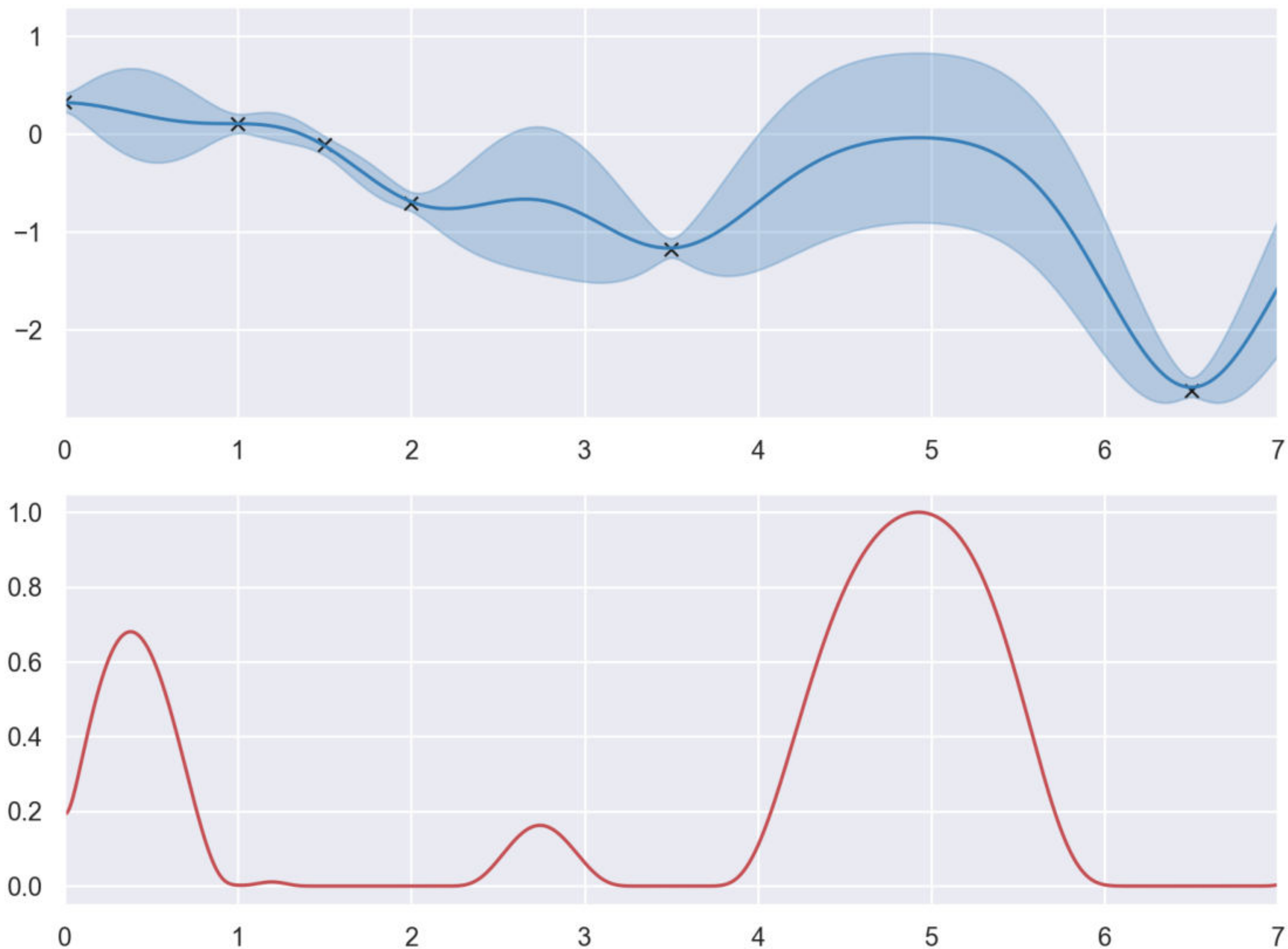


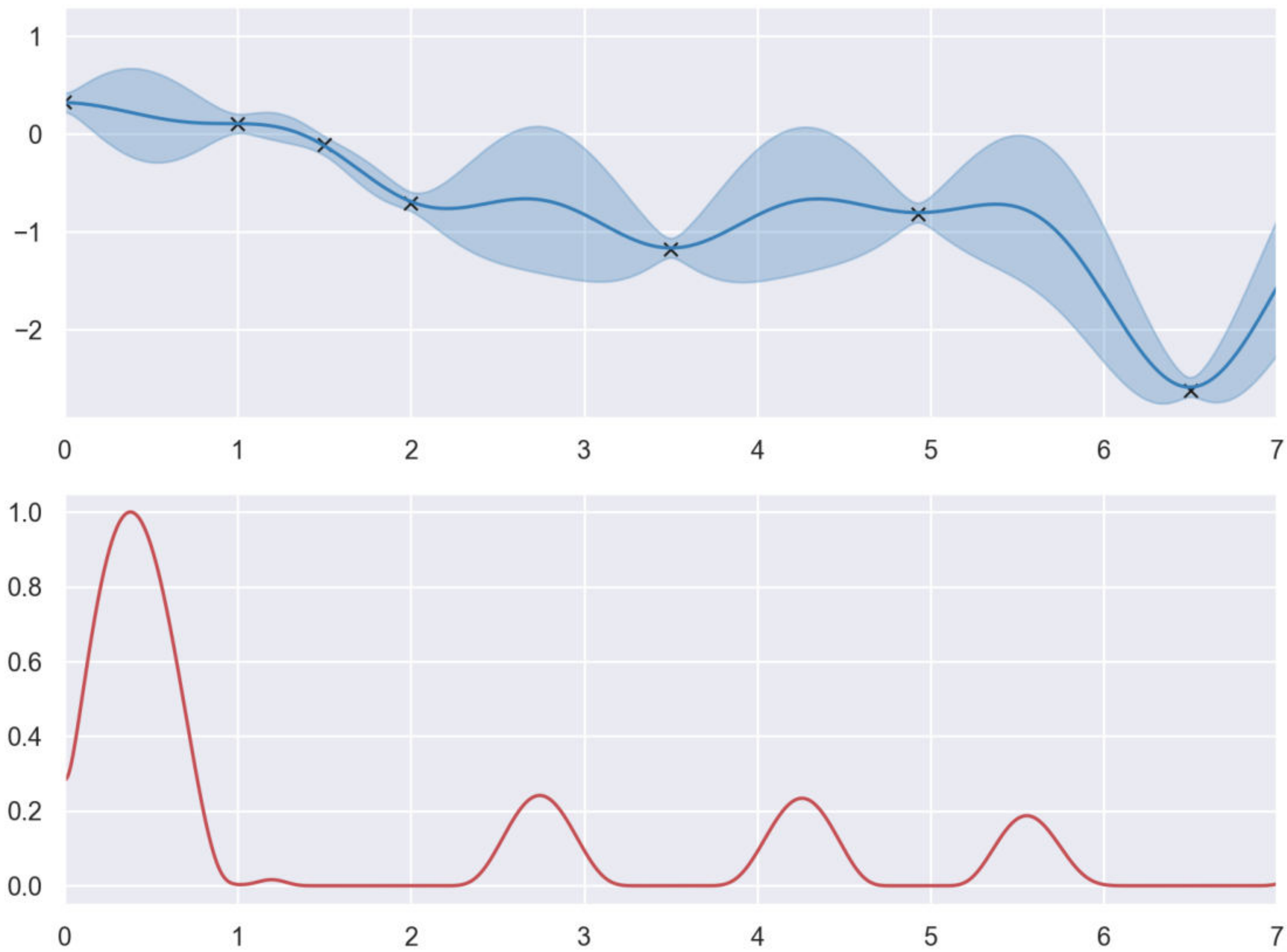
1
0
−1
−2
0
1
2
3
4
5
6
7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0

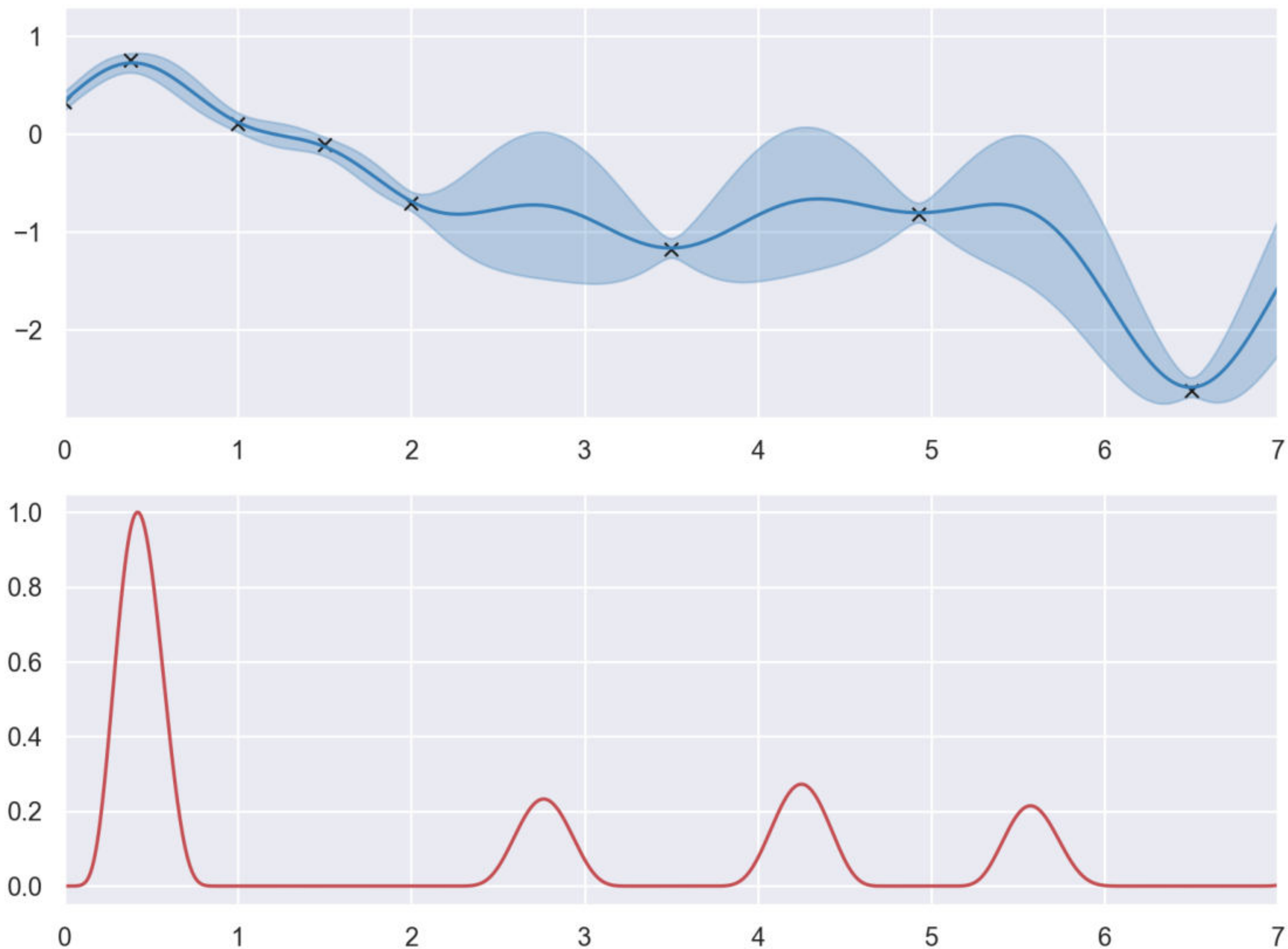


1
0
−1
−2
0
1
2
3
4
5
6
7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0

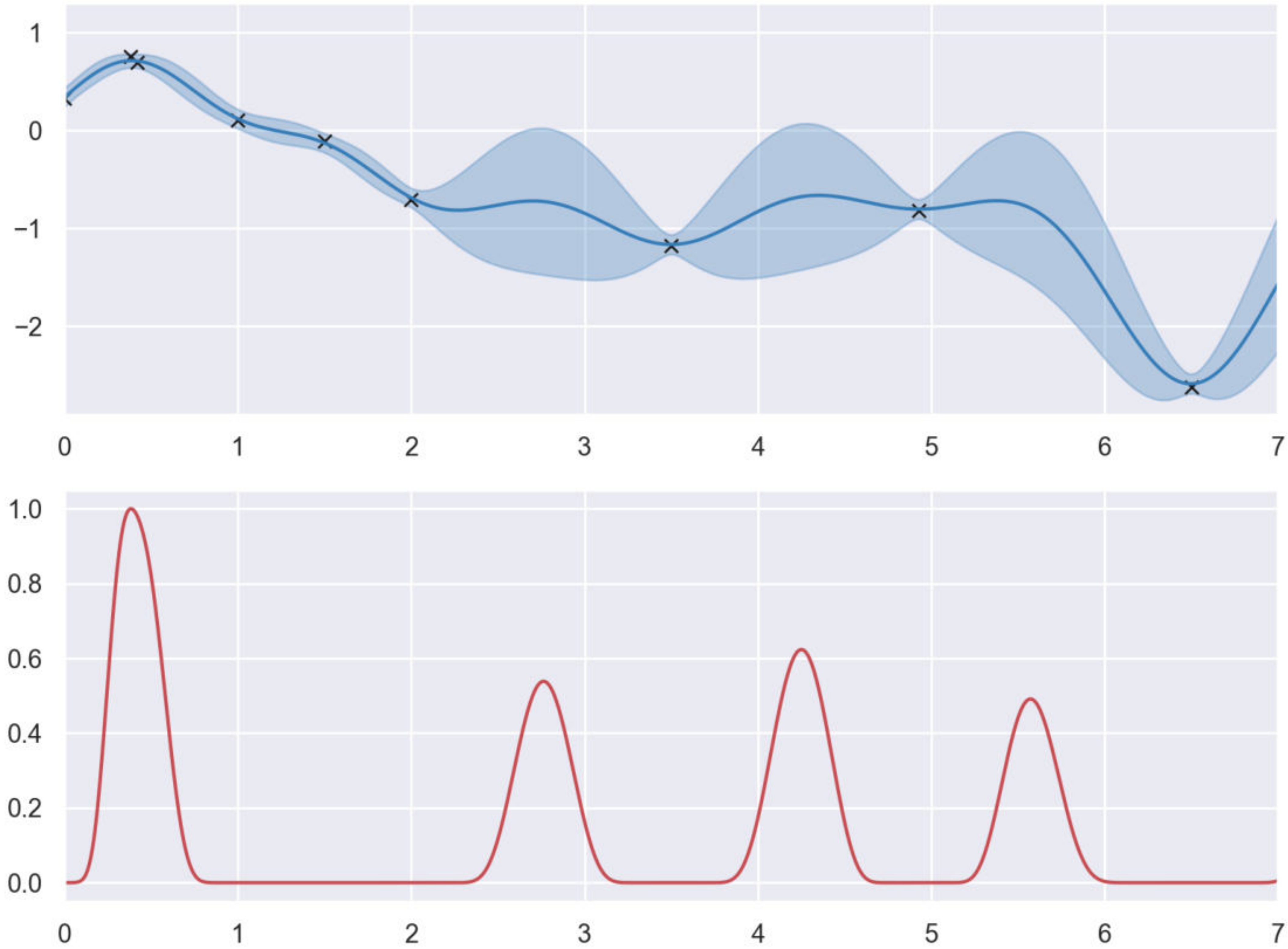

1
0
−1
−2
0
1
2
3
4
5
6
7
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
5
6
7

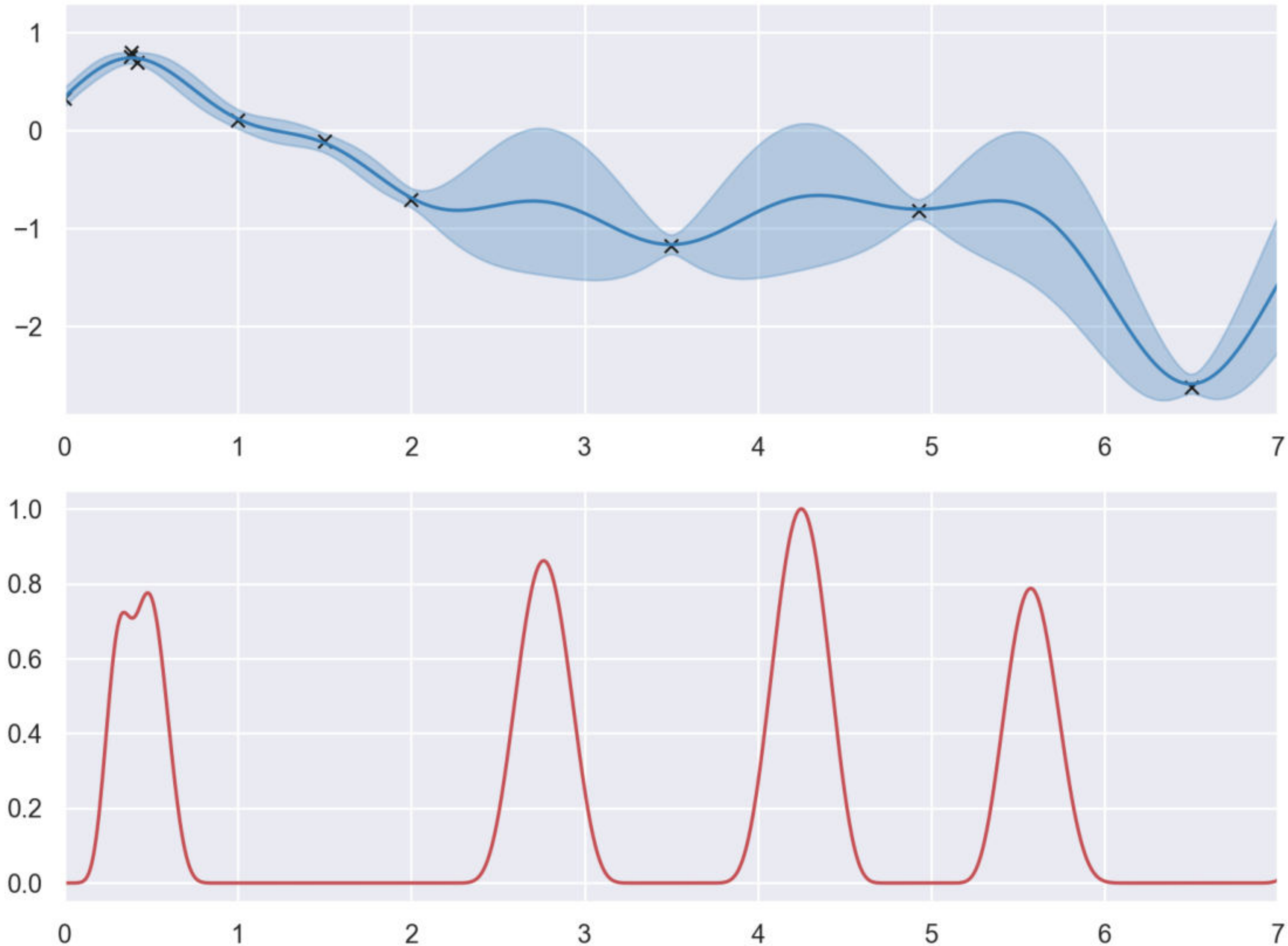

1
0
−1
−2
0 1 2 3 4 5 6 7
1.0
0.8
0.6
0.4
0.2
0.0
0 1 2 3 4 5 6 7


1
0
−1
−2
0
1
2
3
4
5
6
7
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
5
6
7

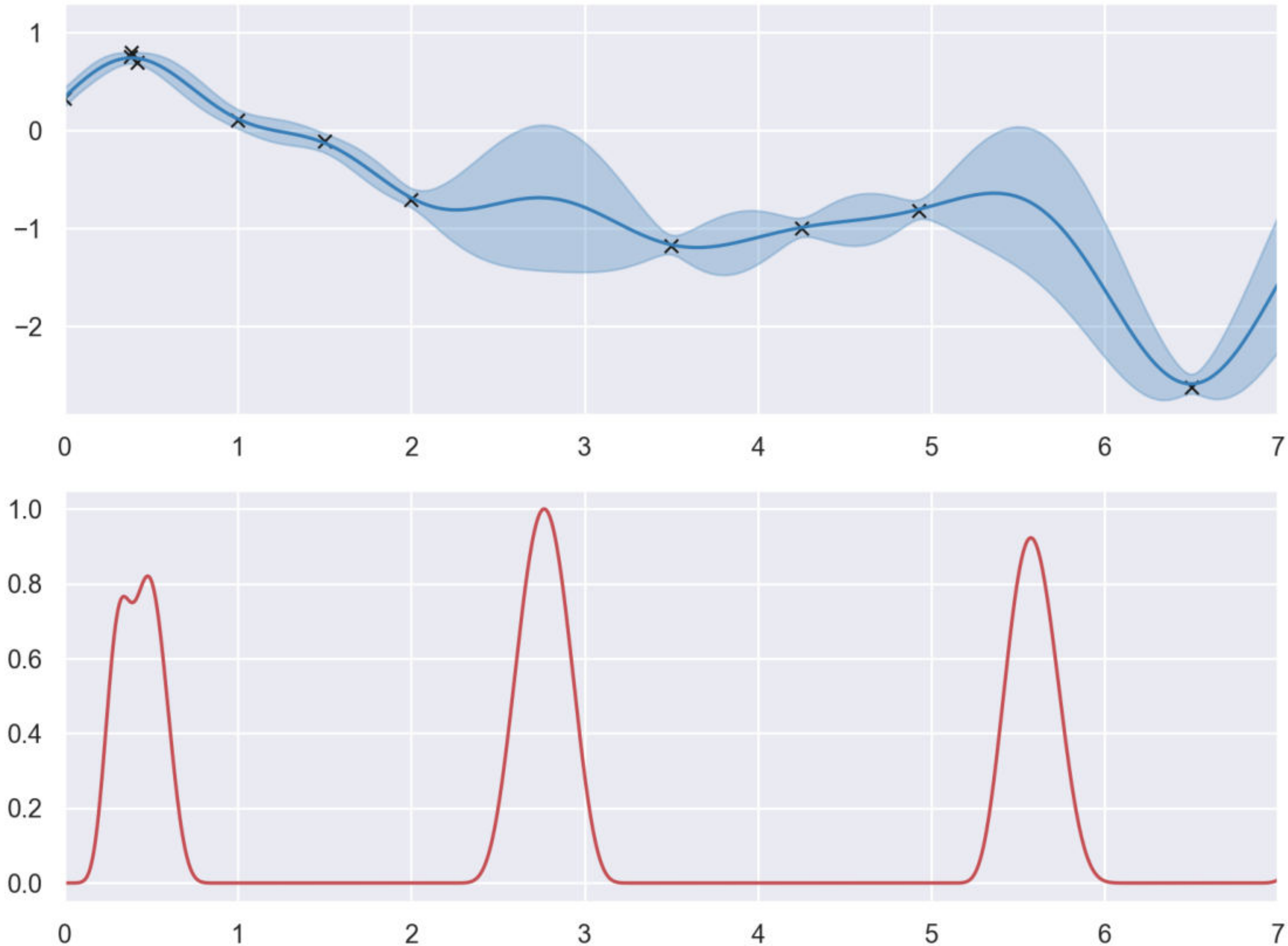

1
0
−1
−2
0
1
2
3
4
5
6
7
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
5
6
7

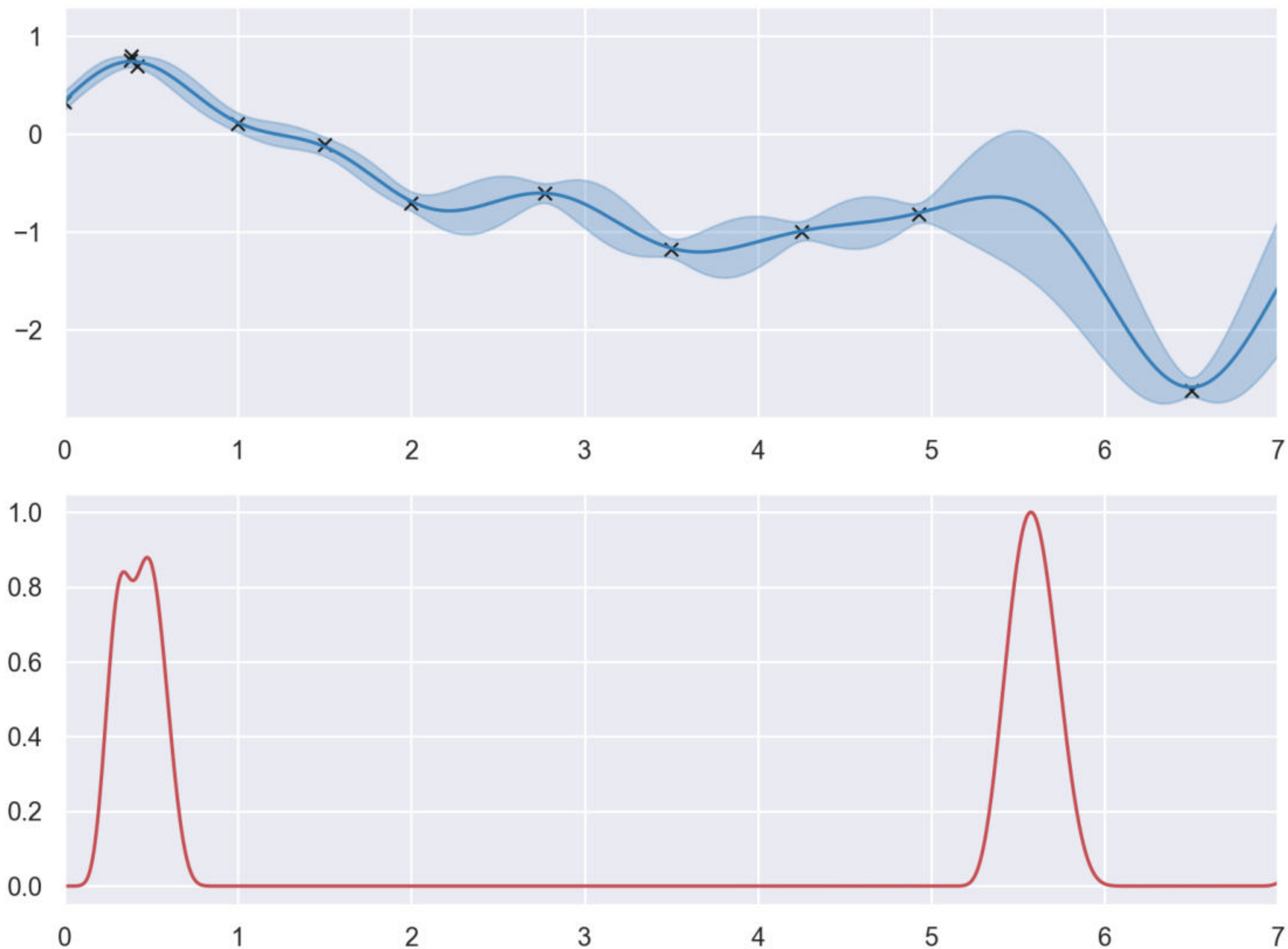
University of Glasgow

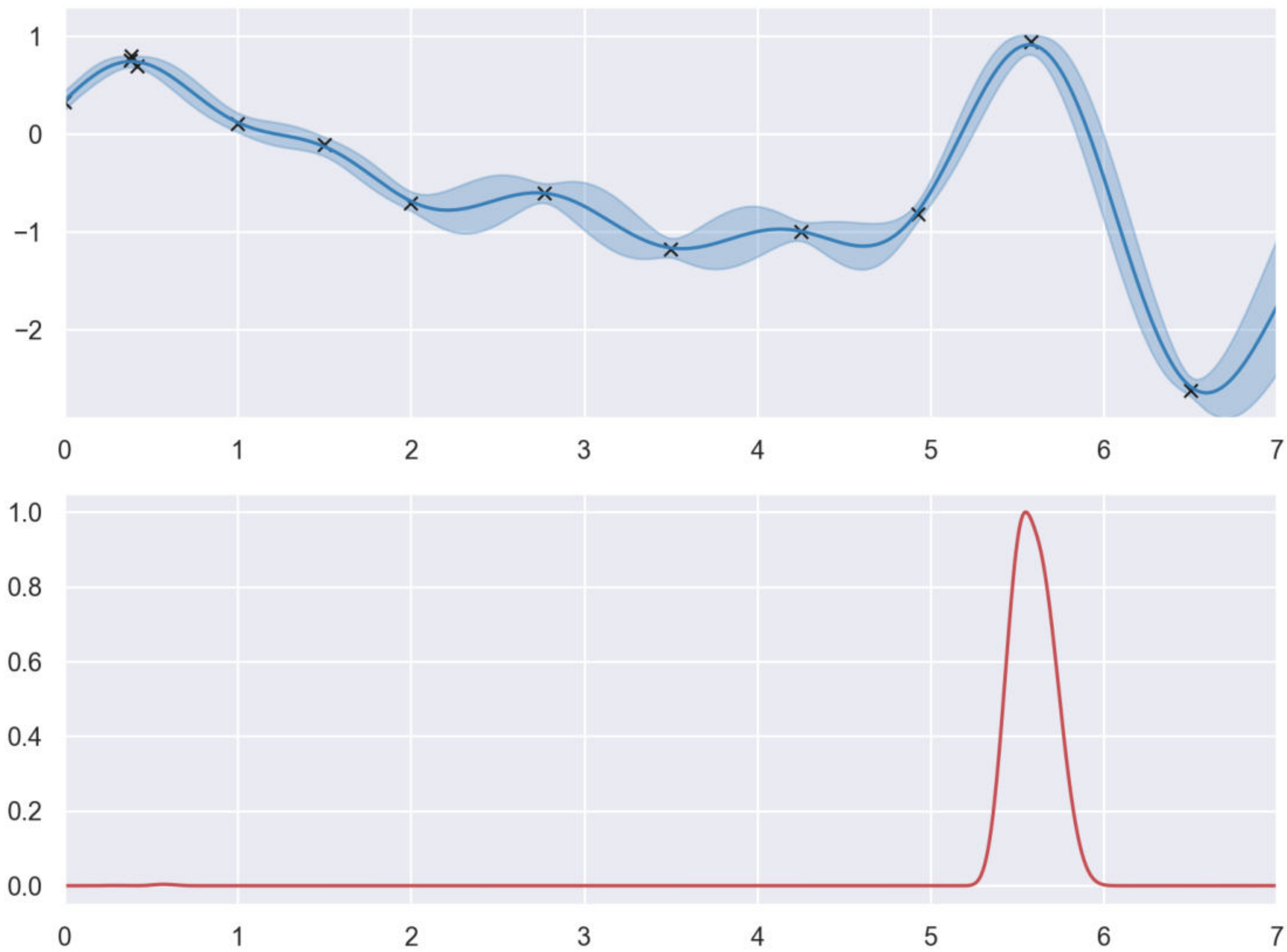

1
0
−1
−2
0
1
2
3
4
5
6
7
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
5
6
7



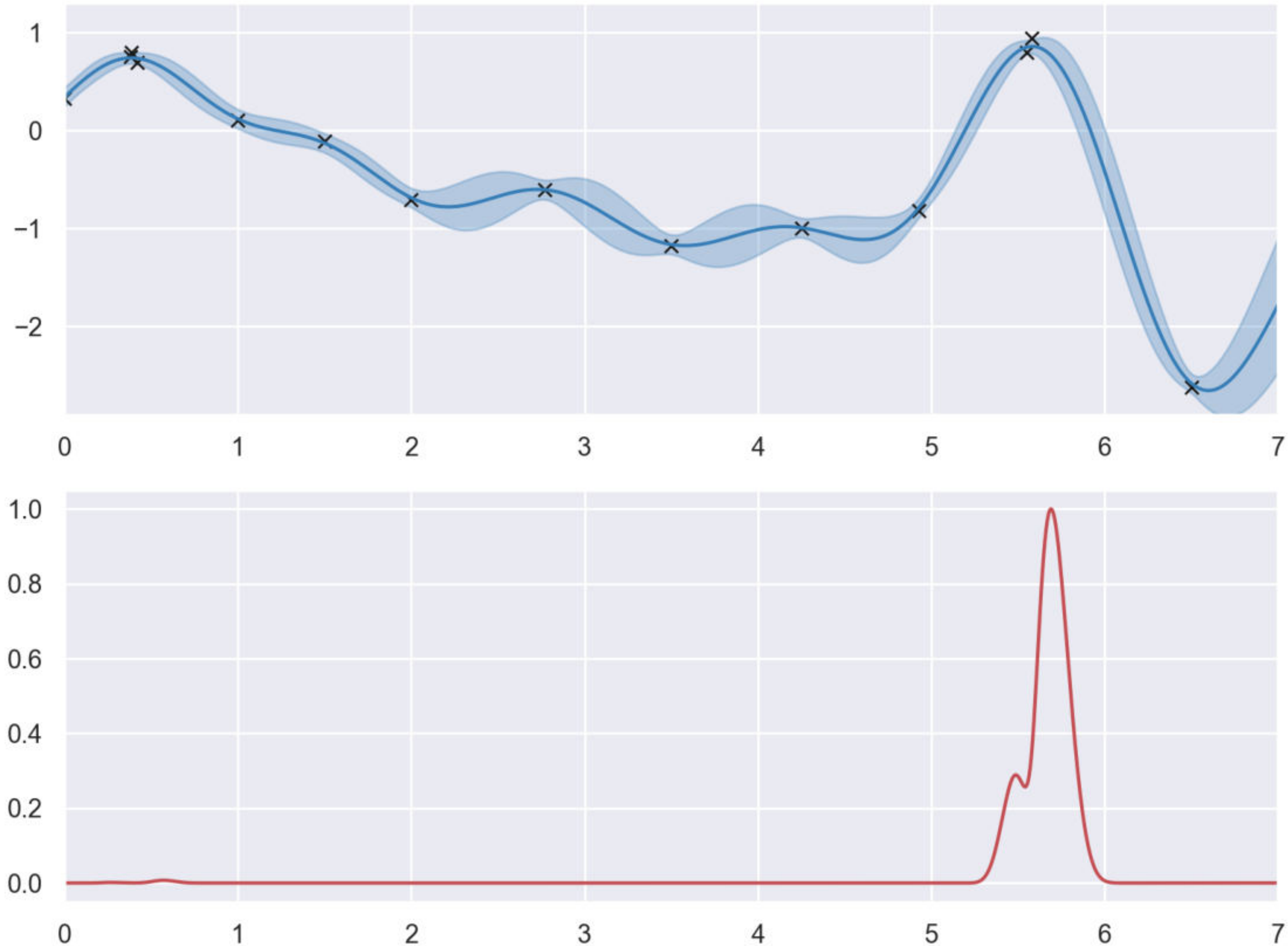
1
0
−1
−2
0
1
2
3
4
5
6
7
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
5
6
7


1
0
−1
−2
0
1
2
3
4
5
6
7
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
5
6
7

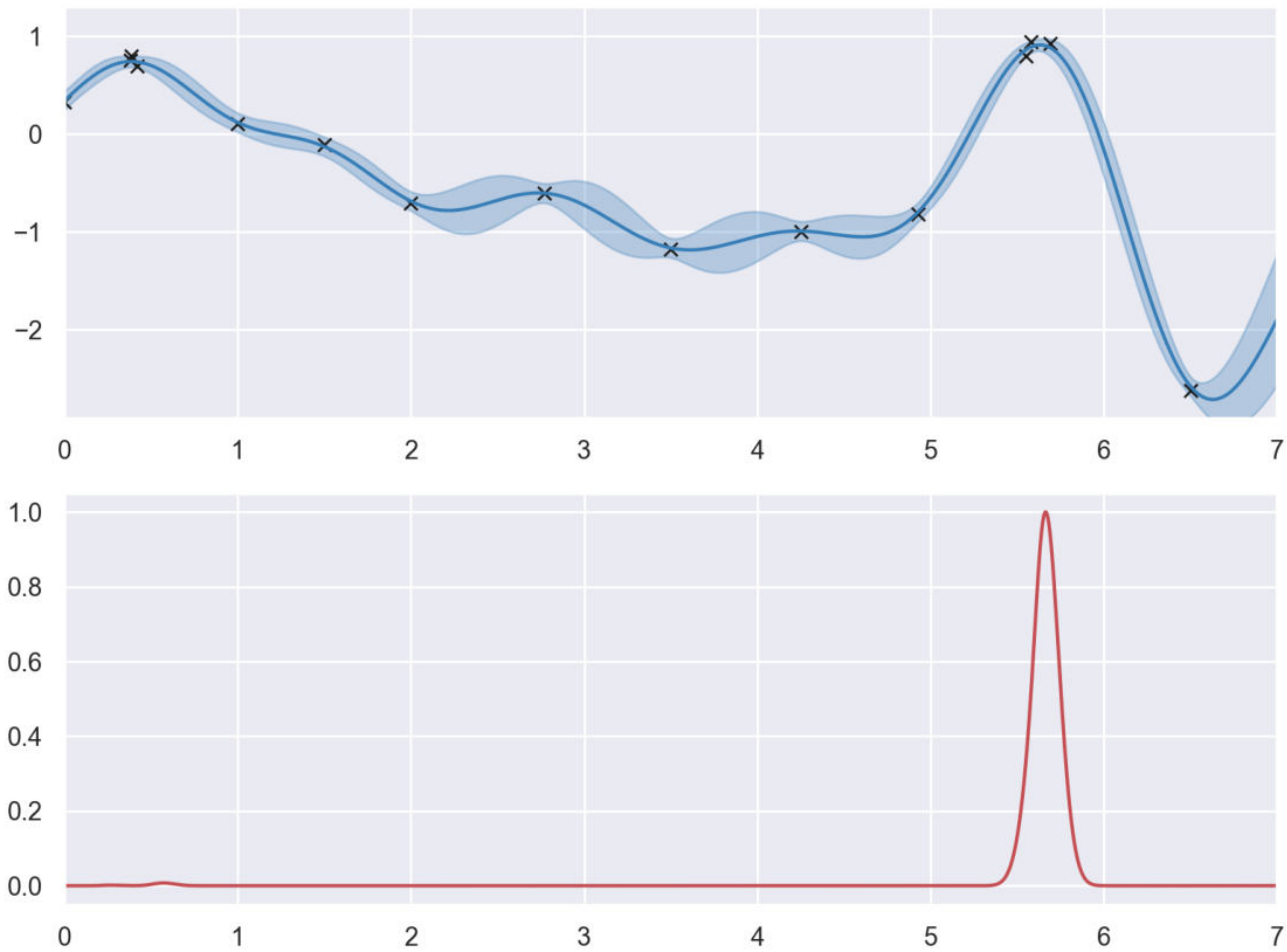

1
0
−1
−2
0
1
2
3
4
5
6
7
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
5
6
7



1
0
−1
−2
0
1
2
3
4
5
6
7
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
5
6
7



1
0
−1
−2
0
1
2
3
4
5
6
7
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
5
6
7



1
0
−1
−2
0
1
2
3
4
5
6
7
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
5
6
7

Efficiency of acquisition function optimisation:

- Querying the surrogate model is (relatively) cheap.
- If the prediction and acquisition function are both differentiable, we can employ numerical optimisation.

$$\frac{d\,\alpha(x)}{d\,x} = \frac{d\,\alpha(x)}{d\,(\mu,\sigma)} \cdot \frac{d\,(\mu,\sigma)}{d\,x}$$

Expected improvement:

$$\begin{aligned}\alpha_{EI}(x; f^\star) &= \int_{f^\star}^{\infty} f \cdot p(f \mid x, \mathcal{D})\, df \\ &= (\mu(x) - f^\star)(1 - \Phi(f^\star; \mu(x), \sigma^2(x)) \\ &\quad + \sigma(x)\phi(f^\star; \mu(x), \sigma^2(x))\end{aligned}$$

Bayesian optimisation algorithm:

1. Pick initial inputs and evaluate blackbox function
2. Fit surrogate model
3. Maximise acquisition function
4. Evaluate blackbox function at maximal input
5. Terminate or go to 2

Software

- GPyOpt [sheffieldml.github.io/GPyOpt]
- Emukit [amzn.github.io/emukit]
- BoTorch [botorch.org]
- GPFlowOpt [github.com/GPflow/GPflowOpt]

Contact

a.uhrenholt.1@research.gla.ac.uk (Anders Kirk Uhrenholt)

bjorn.jensen@glasgow.ac.uk (Dr. Bjørn Sand Jensen)