Investigating Effective Subspaces via Neural Network Training for Bayesian Optimization

Po-Kai Chen (s4283341), Max Jacobs (s4225414)

Abstract

High-dimensional Bayesian Optimization (BO) often assumes that the objective function varies primarily within a lower-dimensional effective subspace, as posited in methods like Random EMbedding Bayesian Optimization (REMBO) [4]. Building upon this assumption, we propose a novel method that integrates neural network training with orthogonal projection matrices to identify such subspaces. By leveraging differentiable surrogate models, our approach approximates the influential directions in the input space, enabling efficient optimization. Experiments on synthetic functions demonstrate that while our method reduces dimensionality, it does not always identify the most relevant subspaces, particularly when no redundant dimensions exist. Benchmark comparisons indicate that our approach performs comparably to PCA-assisted Bayesian Optimization (PCA-BO) in most cases, with reduced variance and increased computational efficiency. These findings suggest that while our method holds promise, further refinement is needed to enhance its accuracy in identifying effective subspaces.

1 Introduction

High-dimensional Bayesian Optimization (BO) often assumes that the objective function varies meaningfully only within a lower-dimensional subspace, known as the effective subspace. This assumption underlies methods like Random EMbedding Bayesian Optimization (REMBO) [4], which utilize random projections to identify such subspaces. Building upon this premise, our study introduces a novel approach that integrates neural network training with orthogonal projection matrices to identify effective subspaces. By leveraging differentiable surrogate models, our method approximates the directions in the input space that significantly influence the function's variation, enabling efficient optimization within a reduced-dimensional space. We evaluate the efficacy of our approach through experiments on synthetic functions and benchmark comparisons, aiming to assess its ability to accurately identify effective subspaces and enhance optimization performance.

2 Theoretical Background

We aim to learn a projection matrix P that projects a high-dimensional vector onto a lower-dimensional space. The following properties of projection matrices will be helpful for constructing P:

1. **Idempotence**: A matrix P is a projection matrix if and only if it satisfies the idempotent property, $P^2 = P$.
2. **Spectral Decomposition**: Since P is idempotent, it is also a normal matrix. By the Spectral Theorem, P can be decomposed as $P = UDU^T$, where U is a unitary matrix and D is a diagonal matrix.
3. **Idempotent Condition on D**: Using the idempotent property, we have:
$$P^2 = P \implies (UDU^T)^2 = UDU^T \implies UD^2U^T = UDU^T \implies D^2 = D.$$
4. **Diagonal Matrix Structure**: From $D^2 = D$, we deduce that D is a diagonal matrix whose diagonal entries are either 1 or 0.

Using point 3 above, we can construct a projection matrix P by specifying a unitary matrix U and a diagonal matrix D with 1's and 0's on its diagonal. This construction serves as the foundation for the algorithm proposed in this work.

3 Proposed Method

The proposed approach trains a projection matrix P within a neural network framework to identify an effective subspace. Given that the function f is a black-box without accessible gradient information, we approximate it using a differentiable surrogate model s trained on sampled data from the search space. The surrogate model is designed to provide gradient information, enabling efficient optimization of the projection matrix.

The training process involves the following steps:

1. **Sample Generation:** Randomly sample a set of vectors $\mathbf{x}$ from the search space and a scalar k from a predefined range.

2. **Projection Computation:** Construct the projection matrix $P = U\,\mathrm{diag}(\mathbf{d})U^\top$ and its complement $I - P$.

3. **Surrogate Evaluation:** Compute the surrogate predictions $s(P\mathbf{x})$ and $s(P\mathbf{x} + k(I - P)\mathbf{x})$, where $P\mathbf{x}$ is the projection onto the subspace and $P\mathbf{x} + k(I - P)\mathbf{x}$ includes a perturbation along the orthogonal complement.

4. **Loss Function Definition:** The loss function is defined as the Mean Squared Error (MSE) between the surrogate model's predictions:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(s(P\mathbf{x}_i) - s(P\mathbf{x}_i + k(I - P)\mathbf{x}_i)\right)^2,$$

   where n is the number of sampled points.

5. **Parameter Updates:** The parameters of U (the orthogonal matrix) and $\mathbf{d}$ (the entries of the diagonal matrix D) are updated in each iteration using gradient information derived from the loss function.

This framework effectively leverages the surrogate model to train the projection matrix and identify the effective subspace, despite the black-box nature of f. Additionally, the diagonal matrix D provides an estimation of the effective dimensionality, as the number of non-zero entries in D corresponds to the dimensions that contribute meaningfully to the function. Algorithm 1 summarizes our method.

Algorithm 1 Training Projection Matrix for Effective Subspace Identification

Train differentiable surrogate model s.
Initialize projection parameters: orthogonal matrix U and diagonal entries d.
for each iteration **do**
    Generate random samples $\mathbf{x}$ from the search space.
    Compute projection matrix P and its complement.
    Compute loss L using surrogate predictions.
    Update U and d based on L.
end for
Output: Trained projection matrix $P = U\,\mathrm{diag}(d)U^\top$.

4 Experimentation and Results

Our experiments consist of two parts: validating our method on synthetic functions and benchmarking its performance in Bayesian Optimization (BO) tasks.

4.1 Experiment 1: Validation with Synthetic Functions

In the first experiment, we test whether our method can identify the effective subspace of a synthetic function. The random functions are generated by constructing a low-dimensional subspace embedded in a higher-dimensional space using an orthonormal basis. A random symbolic equation is defined over this subspace, and high-dimensional inputs are projected onto it for evaluation, ensuring that the function varies meaningfully only within the effective subspace. The evaluation is based on the found dimensions and the mean principal angles [2] between the identified subspace and the true effective subspace.

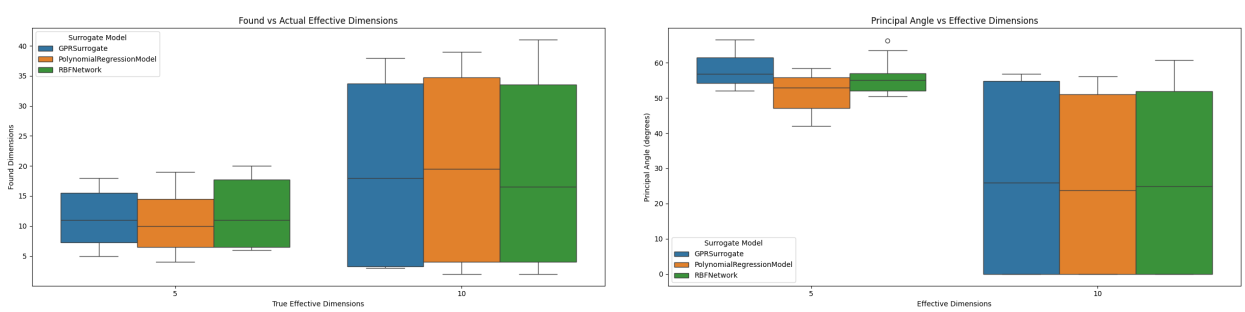
The results, as shown in Table1 and Figure1, reveal several key findings. First, the principal angles are not low in these problems, suggesting that although our method successfully reduces dimensionality, it does not necessarily identify the relevant subspace. Additionally, the method underestimates the dimensionality when there is no redundant dimensionality, as observed in the 10/10 case.

Second, we observed that the MSE loss is sometimes zero throughout the training. Upon further investigation, we found that this occurs because the surrogate model predicts the same value (often zero) for both the sampled points and the perturbed points. This indicates that our method is highly dependent on the accuracy of the surrogate model, and its failure to find a close enough subspace may stem from these inaccuracies.

Lastly, the differences in performance among the three surrogate models are minimal and can be ignored. This suggests that none of the surrogate models are sufficiently accurate for the task, and a more tailored surrogate approach may be necessary to improve the ability of our method to effectively identify the true effective subspace.

Table 1: Experimentation Results: Found Dimensions and Principal Angles

Surrogate	**Total Dim**	**Effective Dim**	**Found Dim Mean**	**Angle Mean**
GPR	10	10	3.6	0.0062
	20	5	6.8	57.8203
	50	5	15.6	58.5172
	100	10	34.8	54.6405
Polynomial	10	10	3.6	0.0105
	20	5	6.2	50.0732
	50	5	15.4	52.5223
	100	10	35.4	51.4750
RBF	10	10	3.8	0.0090
	20	5	7.2	56.6226
	50	5	17.0	55.4126
	100	10	34.2	54.5662



((a)) Found vs. Actual Effective Dimensions.

((b)) Principal Angle vs. Effective Dimensions.

Figure 1: Comparison of performance across different surrogate models.

4.2 Experiment 2: Benchmarking on Bayesian Optimization

In our second experiment, we benchmarked our method against PCA-assisted Bayesian Optimization (PCA-BO) [3], which employs linear transformations to reduce dimensionality. Both approaches aim to enhance optimization efficiency by identifying and operating within a lower-dimensional subspace.

We conducted experiments on problem instances from the Black-Box Optimization Benchmarking (BBOB) suite [1], which provides a comprehensive set of 24 noiseless, single-objective, and scalable test functions designed to evaluate optimization algorithms. The specific configurations were:

- **Function IDs (fid):** [1, 8, 12, 15, 21]
- **Dimensions:** [2, 10, 40]
- **Instance IDs (iid):** [0, 1, 2]

Each method was allocated a total budget of dimension $\times$ 10 function evaluations for the optimization process. In our approach, we used a fixed budget of 50 evaluations for subspace construction, followed by dimension $\times$ 10 evaluations for optimization within the identified subspace.

According to the results shown in Figure 2, our approach performs comparably to PCA-BO on the first four functions (f1, f8, f12, and f15). However, for f21 (representing the fifth BBOB category, multi-modal functions with weak global structure), our method underperforms significantly. This result suggests either that this category does not have a meaningful subspace or that our method failed to identify one effectively. In Experiment 1 we showed that this method can be heavily reliant on the surrogate model, and with complex functions like these the surrogate model poorly captures the behavior of the landscape.

However, our method exhibits a smaller variance in the solutions found across different runs, indicating more consistent performance. Furthermore, it is computationally more efficient than PCA-BO, running all experiments with our method took less than an hour, whereas PCA-BO required almost a day. These findings highlight the potential of our approach for scenarios where computational efficiency and consistency are critical.

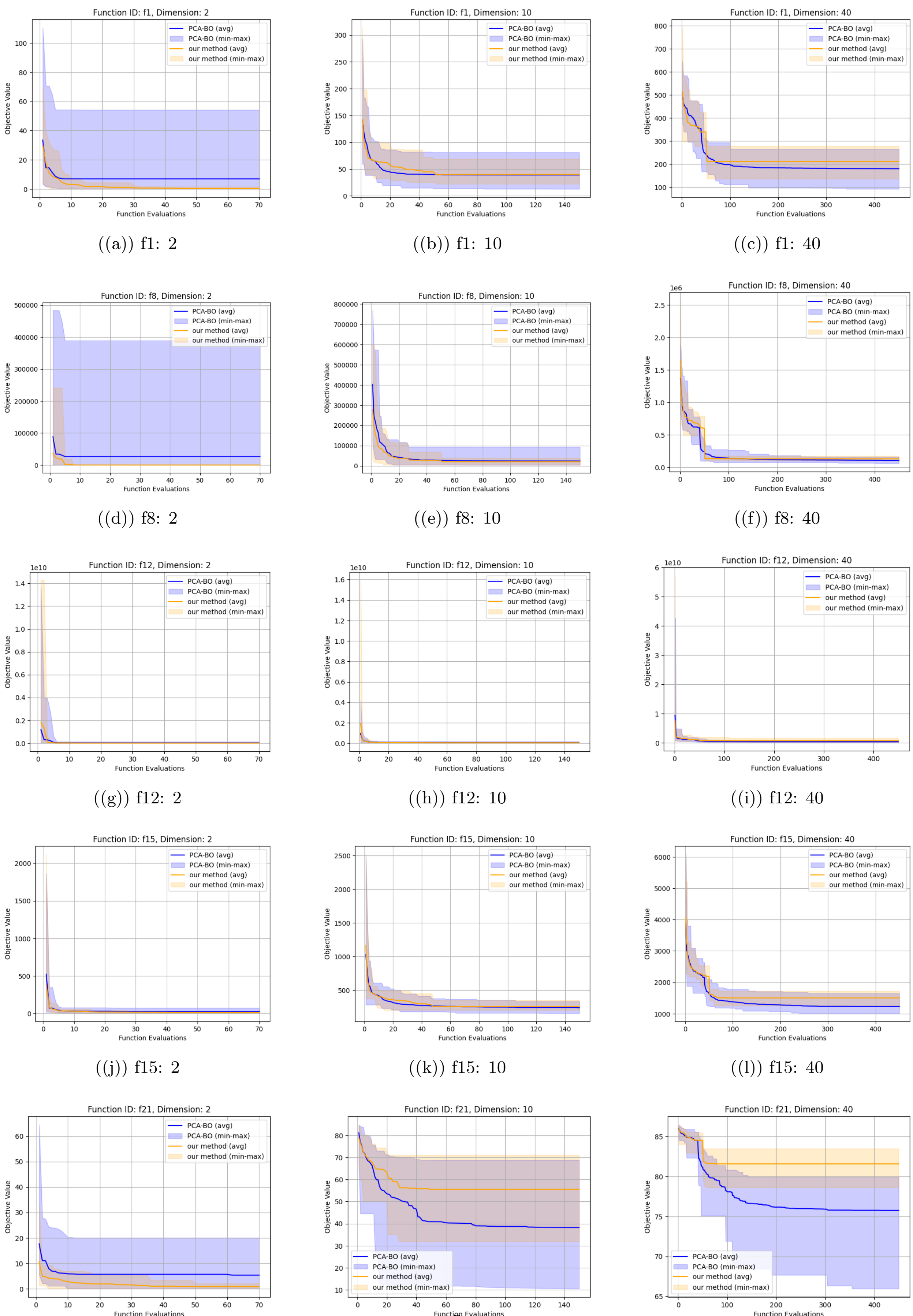
5 Conclusion

In this study, we introduced a novel method for identifying effective subspaces in high-dimensional optimization problems, building upon the assumption made in REMBO [4] that such subspaces exist. Our approach integrates neural network training with orthogonal projection matrices to detect these subspaces, thereby enhancing the efficiency of Bayesian Optimization (BO) processes.

Our experiments yielded several key insights:

- **Synthetic Function Analysis:** The method successfully reduced dimensionality but did not consistently identify the most relevant subspaces. This limitation is attributed to the surrogate models' accuracy, which is crucial for the method's success. Instances where the Mean Squared Error (MSE) loss remained at zero throughout training indicated that surrogate models predicted identical values for both sampled and perturbed points, undermining the method's effectiveness.
- **Benchmarking with PCA-BO:** When compared to PCA-assisted Bayesian Optimization (PCA-BO), our method demonstrated comparable performance across various function categories, with the exception of the fifth BBOB category, where it underperformed. This discrepancy suggests that either the function in this category lacked meaningful subspaces or our method was unable to identify them effectively. However, our method ran much faster than PCA-BO, completing all experiments in under an hour compared to PCA-BO's nearly day-long duration. Additionally, our approach typically had less variance in solutions across different runs.

These findings show the viability of our method in scenarios where the assumption of an existing effective subspace holds. However, the dependence on the accuracy of the surrogate model highlights a significant limitation. Future research should focus on developing customized surrogate modeling techniques to improve the identification of effective subspaces. Additionally, exploring the method's applicability to a broader range of functions, particularly those without clear subspace structures, will be essential in assessing and improving its generalizability.



((a)) f1: 2 ((b)) f1: 10 ((c)) f1: 40

((d)) f8: 2 ((e)) f8: 10 ((f)) f8: 40

((g)) f12: 2 ((h)) f12: 10 ((i)) f12: 40

((j)) f15: 2 ((k)) f15: 10 ((l)) f15: 40

((m)) f21: 2 ((n)) f21: 10 ((o)) f21: 40

Figure 2: Comparison of f1, f8, f12, f15, and f21 at 2, 10, and 40.

References

[1] Nikolaus Hansen, Anne Auger, Steffen Finck, and Raymond Ros. The black-box optimization benchmarking (bbob) test suite. *arXiv preprint arXiv:2007.00363*, 2020.

[2] Andrew V Knyazev and Peizhen Zhu. Angles between subspaces and their tangents. *arXiv preprint arXiv:1209.0523*, 2012.

[3] Elena Raponi, Hao Wang, Mariusz Bujny, Simonetta Boria, and Carola Doerr. High dimensional bayesian optimization assisted by principal component analysis. *arXiv preprint arXiv:2007.00925*, 2020.

[4] Ziyu Wang, Frank Hutter, Masrour Zoghi, David Matheson, and Nando de Freitas. Bayesian optimization in a billion dimensions via random embeddings. *Journal of Artificial Intelligence Research*, 55:361–387, 2016.