An Evaluation of Zero-Cost Proxies - from Neural Architecture Performance Prediction to Model Robustness

Jovita Lukasik[1], Michael Moeller[1], and Margret Keuper[1,2]

[1] University of Siegen
{jovita.lukasik,michael.moeller,margret.keuper}@uni-siegen.de
[2] Max Planck Institute for Informatics, Saarland Informatics Campus

Abstract. Zero-cost proxies are nowadays frequently studied and used to search for neural architectures. They show an impressive ability to predict the performance of architectures by making use of their untrained weights. These techniques allow for immense search speed-ups. So far the joint search for well-performing and robust architectures has received much less attention in the field of NAS. Therefore, the main focus of zero-cost proxies is the clean accuracy of architectures, whereas the model robustness should play an evenly important part. In this paper, we analyze the ability of common zero-cost proxies to serve as performance predictors for robustness in the popular NAS-Bench-201 search space. We are interested in the single prediction task for robustness and the joint multi-objective of clean and robust accuracy. We further analyze the feature importance of the proxies and show that predicting the robustness makes the prediction task from existing zero-cost proxies more challenging. As a result, the joint consideration of several proxies becomes necessary to predict a model's robustness while the clean accuracy can be regressed from a single such feature.

Keywords: neural architecture search · zero-cost proxies · robustness.

1 Introduction

Neural Architecture Search (NAS) [7,34] seeks to automate the process of designing high-performing neural networks. This research field has gained immense popularity in the last years, with only a few NAS papers in 2016 to almost 700 in 2022 [34]. The ability of NAS to find high-performing architectures, capable of outperforming hand-designed ones, notably on image classification [39], makes this research field highly valuable. However, classical approaches like reinforcement learning [39,18] or evolutionary algorithms [26] are expensive, which led to a focus-shift towards improving the search efficiency. Therefore different approaches such as one-shot methods [20] or performance prediction methods [35] were introduced. Recently, zero-cost proxies (ZCP) [23], as part of performance prediction strategies, were developed. These proxies build upon fast computations, mostly in one single forward and backward pass on an untrained model,

and attempt to predict the accuracy that the underlying architecture will have after training. In recent years, several zero-cost proxies were introduced [23,1], including simple architectural baselines as FLOPS or the number of parameters. NAS-Bench-Suite-Zero [14] provides a more in-depth analysis and evaluates 13 different ZCPs on 28 different tasks to show the effectiveness as a performance prediction technique.

So far, the main focus of NAS research was the resulting performance of the architectures on a downstream task. Another important aspect of networks, namely their robustness, has been less addressed in NAS so far. Most works targeting both high accuracy and a robust network rely on one-shot methods [12,24]. However, using only ZCPs as a performance prediction model for the multi-objective has not been addressed. The search for architectures that are robust against adversarial attacks is especially important for computer vision and image classification since networks can be easily fooled when the input data is changed using slight perturbations that are even invisible to the human eye. This can lead to networks making false predictions with high confidence.

This aspect is particularly important in the context of NAS and ZCPs because the search for robust architectures is significantly more expensive: an the architecture's robustness is to be evaluated on trained architectures against different adversarial attacks [8,16,4].

In this paper, we therefore address the question: How transferable are the accuracy-focused ZCPs to the architecture's robustness? A high-performing architecture is not necessarily robust. Therefore, we analyze which ZCPs perform well for predicting clean accuracy, which are good at predicting robustness, and which do well at both. For our evaluation, we leverage the recently published robustness dataset [13] which allows for easily accessible robustness evaluations on an established NAS search space [6]. Since every ZCP provides a low-dimensional (scalar) measure per architecture, we understand each ZCP as a feature dimension in a concatenated feature vector and employ random forest regression as a predictor of clean and robust accuracy. This facilitates not only the evaluation of the performance or correlation of the different measures with the prediction target but also to gain direct access to every proxy's feature importance compared with all others.

Our evaluation of all ZCPs from NAS-Bench-Suite-Zero allows us to make the following observations:

- While the correlation of every single ZCP with the target is not very strong, the random forest regression allows predicting the clean accuracy with very good and the robust accuracy with good precision.
- As a result, we expect that leveraging several ZCPs simultaneously will allow for NAS with the aim to find robust architectures, while every single ZCP will likely not provide the needed information.
- When analyzing the feature importance, ZCPs using Jacobian-based information generally carry the most employed information.
- The analysis of the feature importance for robust model accuracy prediction confirms that various ZCPs contribute strongly to the overall prediction.

2   Related Work

2.1   Zero-Cost Proxies for NAS

NAS is the process to automate the design of neural architectures with the goal to find a high-performing architecture on a particular dataset. In the last few years, this research field gained immense popularity and is able to surpass hand-designed neural architectures on different tasks, especially on image classification. See [34] for a survey. For fast search and evaluation of found architectures, many NAS methods make use of performance prediction techniques via surrogate models. The surrogate model predicts the performance of an architecture, without the need of training the architecture, with the goal to keep the query amount (i.e. the amount necessary training data to train the surrogate model) low. Each query means one full training of the architecture, therefore, successful performance prediction-based NAS methods are able to use only a few queries. [32,27,33,36,22] show improved results using surrogate models keeping the query amount low. However, these methods use the validation or test accuracy of the architecture as a target and require high computation time. In order to predict the performance of an architecture without full training, zero-cost proxies (ZCPs), measured on untrained architectures, can be used [23]. The idea is that these ZCPs, which often require only one forward and one backward pass on a single mini-batch, are somehow correlated with the resulting performance of the architecture after full training. [23] originally used ZCPs for NAS by analyzing the linear regions and how well they are separated. In contrast, [1] uses pruning-at-initialization techniques ([17,31,29]) as ZCPs. The best performing ZCP in [1] `synflow` is data-independent, which does not consider the input data for the proxy calculation. Another data-independent architecture was proposed by [19]. Other approaches use the neural tangent kernel for faster architecture search [2].

The recent benchmark NAS-Bench-suite zero [14] compares different ZCPs on different NAS search spaces and shows how they can be integrated into different NAS frameworks. Other works include ZCPs into Bayesian optimization NAS approaches (as for example [28,35]) and one-shot architecture search [37]. In contrast, this paper evaluates how well ZCPs can predict the robust accuracy of a model under adversarial attacks, and demonstrates (surprising) success in combining different ZCP features in a random-forest classifier.

2.2   Robustness in NAS

Compared to searching for an architecture with the single objective of having a high performance, including the architecture's robustness results in an even more challenging task that requires a multi-objective search. Recent works that search for both high-performing and robust architectures combine both objectives in one-shot search approaches [9,5,24,12]. [5] includes a parameter constraint in the supernet training in order to reduce the Lipschitz constant. Also [12] adds additional maximization objectives, the certified lower bound and Jacobian norm bound, to the supernet training. [24] includes the loss landscape of the Hessian

into the bi-level optimization approach from [20]. In contrast to these additional objectives, [9] proposes adversarial training of the supernet training for increased network robustness. The recent robustness dataset [13] facilitates this research area. All 6 466 unique architectures in the popular NAS cell-based search space, NAS-Bench-201 [6], are evaluated on four different adversarial attacks (FGSM [8], PGD [16], APGD and Square [4]) with different attack strengths on three different image datasets, CIFAR-10 [15], CIFAR-100 [15], and ImageNet16-120 [3].

3   Background on Zero-Cost Proxies

As presented in NAS-Bench-Suite-Zero [14], we can differentiate the proxies into different types: Jacobian-based (★), pruning-based (◆), baseline (■), piecewise-linear (▼), and also Hessian-based (•) zero-cost proxies. In the following, we will provide more information about the ZCPs, which we evaluate in this paper.

3.1   Jacobian-based

Mellor et al. [23] were the first to introduce ZCPs into NAS, by analyzing the network behavior using local linear operations for each input $\mathbf{x}_i \in \mathbb{R}^D$ in the training data mini-batch, which can be computed by the Jacobian of the network for each input. The idea is based on the fact that a resulting well-performing untrained network is supposed to be able to distinguish the local linear operations of different data points. For that, the metric `jacov` was introduced, which is using the correlation matrix of the Jacobian as a Gaussian kernel. The score itself is the Kullback-Leibler divergence between a standard Gaussian and a Gaussian with the mentioned kernel. Therefore, the higher the score, the better the architecture is likely to be.

Building on that, [23] further introduced `nwot` (Neural Architecture Search without Training), which forms binary codes, depending on whether the rectified linear unit is active or not, which define the linear region in the network. Similar binary codes for two input points indicate that it is more challenging to learn to separate these points. [21] developed `nwot` even further by introducing `epe-nas` (Efficient Performance Estimation Without Training for Neural Architecture Search). The goal of epe-nas is to evaluate if an untrained network is able to distinguish two points from different classes and equate points from the same class. This is measured by the correlation of the Jacobian (`jacov`) for input data being from the same class. Therefore, the resulting correlation matrix can be used to analyze how the network behaves for each class and thus may indicate if this network can also distinguish between different classes.

As an alternative to these methods, [1] proposed a simple proxy, `grad-norm`, which is simply the sum of the Euclidean norm of the weight gradients.

So far, these Jacobian-based measurements focused on the correlation with the resulting clean performance of architectures. In this paper, we are also interested in the correlation and influence on the robust accuracy of the architecture. As also used in [13], [12] combined the search for a high-performing architecture

that is also robust against adversarial attacks, by including the Frobenius norm of the Jacobian (`jacob-fro`) into the search. As introduced in [11] the change of the network output, when a perturbed data point $\mathbf{x}_i + \epsilon, \epsilon \in \mathbb{R}^D$ is the input to the network instead of the clean data point $\mathbf{x}_i$, can be used as a measurement for the robustness of the architecture: the larger the change, the more unstable is the network in case of perturbed input data. This change can be measured by the square of the Frobenius norm of the difference of the network's prediction on perturbed and unperturbed data.

3.2   Pruning-based

Pruning-based ZCPs are based on network pruning metrics, which identify the least important parameters in a network at initialization time. [17] introduced a measurement, `snip` (Single-shot Network Pruning), which uses a connection sensitivity metric to approximate the change in the loss, when weights with a small gradient magnitude are pruned. Based on that, [31] improves `snip` by approximating the change in the loss after weight pruning in their `grasp` (Gradient Signal Preservation) metric. Lastly, [29] investigated these two pruning-based metrics in terms of layer collapse, and proposes `synflow`, which multiplies all absolute values of the parameters in the network, which is independent of the training data.

In contrast to that, [30] obtains and aggregates the Fisher information `fisher` for all channels in a convolution block, to identify the channel with the least important effect on the network's loss.

3.3   Piecewise Linear

[19] proposes the `zen` score, motivated the observation that a CNN can be also seen as a composition of piecewise linear functions being conditioned on activation patterns. Therefore, they propose to measure the network's ability to express complex functions by its Gaussian complexity. This score is data-independent since both the network weights and the input data are sampled from a standard Gaussian distribution.

3.4   Hessian-based

As mentioned in Sec. 3.1, the goal for ZCP research was mainly motivated by finding a measurement, which is correlated with the network's performance after training. In this paper, we want to shift the focus towards the robustness of architectures, which is also a crucial aspect of neural architectures. [24] also included the robustness as a target for architecture search, by considering the smoothness of the loss landscape of the architecture. [38] investigated the connection between the smoothness of the loss landscape of a network and its robustness, and shows that the adversarial loss is correlated with the biggest eigenvalue of the Hessian. A small Hessian spectrum implies a flat minimum, whereas a large Hessian spectrum, implies a sharp minimum, which is more sensitive to changes in the input and therefore can be more easily fooled by adversarial attacks.

3.5   Baselines

In addition to the above mentioned developed ZCPs, basic network information have also been successfully used as ZCPs [1,25]. The most common baseline proxies are the number of FLOPS (`flops`) as well as the number of parameters (`params`). In addition to these, [1] also considers the sum of the L2-norm of the untrained network weights, `l2-norm`, and the multiplication of the network weights and its gradients, `plain`.

4   Feature Collection and Evaluation

In the following, we are going to describe our evaluation setting.

4.1   NAS-Bench-201

NAS-Bench-201 [6] is a cell-based search space [34], in which each cell has 4 nodes and 6 edges. The node represents the architecture's feature map and each edge represents one operation from a predefined operation set. This operation set contains 5 different operations: 1×1 convolution, 3×3 convolution, 3×3 avg.pooling, $\mathrm{skip-connection}$, zero. The cells are integrated into a macro-architecture. The overall search space has a size of $5^6 = 15\,625$ different architectures, from which $6\,466$ architectures are unique and non-isomorphic. Isomorphic architectures have the same information flow in the network architecture, resulting in similar outcomes and only differ due to numerical reasons. All architectures are trained on three different image datasets, CIFAR-10 [15], CIFAR-100 [15], and ImageNet16-120 [3].

4.2   Neural Architecture Design and Robustness Dataset

The dataset by [13] evaluated all the unique architectures in NAS-Bench-201 [6] against three white-box attacks, i.e., FGSM [8], PGD [16], APGD [4], as well as one black-box attack, Square Attack [4] to evaluate the adversarial robustness of the architectures, given by the different topologies. In addition, all architectures were also evaluated on corrupted image datasets, CIFAR-10 C and CIFAR-100-C [10]. Along with these evaluations, the dataset also shows three different use cases, on how the data can be used: evaluation of ZCPs for robustness, NAS on robustness, and an analysis of how the topology and the design of an architecture influence its resulting robustness. This dataset also provides evaluations for the Frobenius norm from Sec. 3.1 and the biggest eigenvalue of the Hessian from Sec. 3.4 as two zero-cost proxies for their first use case. Note, the latter proxy, the Hessian, was only evaluated on the CIFAR-10 image data [15].

4.3   Collection

Both datasets [14,13] provide us with the necessary proxies for the NAS-Bench-201 [6] search space evaluations. Therefore we will analyze 15 different proxies

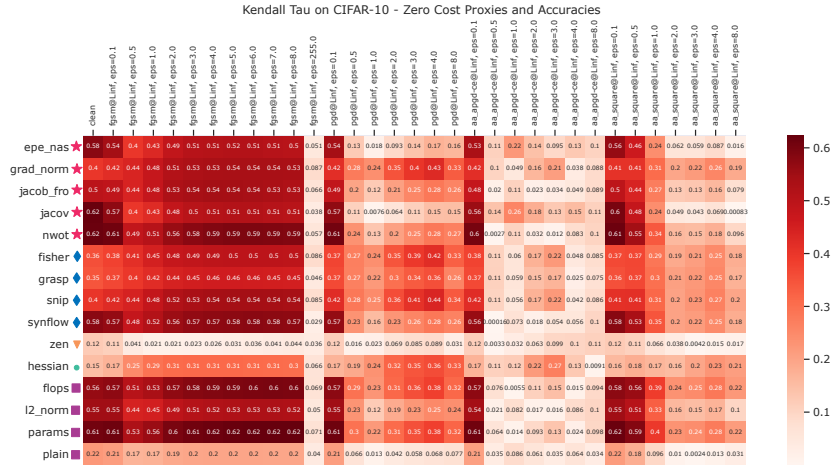Kendall Tau on CIFAR-10 - Zero Cost Proxies and Accuracies

Fig. 1: Kendall tau rank correlation in **absolute values** between all zero-cost proxies computed on all architectures given in the robustness dataset [13] to the test accuracy and adversarial attacks for CIFAR-10.

for 6 466 architectures on three different image datasets. Note, the ZCPs in [14] were evaluated on the validation accuracy, while [13] provide test accuracy information. We will focus here on the test accuracies.

5 Evaluations of Zero-Cost Proxies

As presented in Sec. 4, we can directly gather all proxies. Having these at hand allows us to evaluate the correlation, influence and importance of each proxy not only on the clean test accuracy of the architectures but also on their robust accuracy.

5.1 Correlation

Figure 1 shows the Kendall tau rank correlation in absolute values between each ZCP and all available accuracies, from clean test accuracy to all adversarial attack test accuracies (`fgsm`, `pgd`, `aa-apgd-ce`, `aa-square`) on CIFAR-10. For the correlation plots on CIFAR-100 and ImageNet16-120 we refer to the supplementary material. The Jacobian-based proxies (especially `jacov` and `nwot`) and the baseline proxies show the highest correlation over all datasets, especially for the FGSM attack. Furthermore, the large correlation within the same attack over different strengths of ϵ values stays steady. However, the more difficult the attack gets, from FGSM over PGD to APGD, the larger is the correlations decrease. Interestingly, the `zen` proxy has the lowest correlation with each test accuracy.

Table 1: Test R^2 of the random forest prediction model for both single objective and multi objectives on the clean test accuracy and the robust test accuracy for $\epsilon = 1/255$.

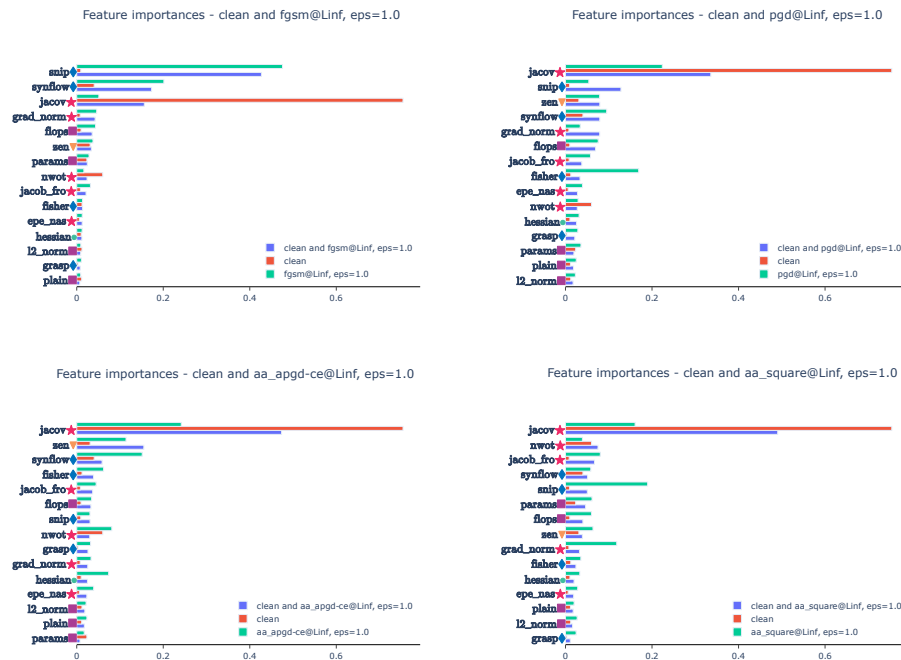| Dataset | Test Accuracy $\epsilon = 1/255$ | | | | | | | | |
| | Single Objective | | | | | Multi Objective | | | |
	Clean	FGSM	PGD	APGD	Squares	Clean-FGSM	Clean-PGD	Clean-APGD	Clean-Squares
CIFAR-10	0.97	0.88	0.65	0.66	0.75	0.92	0.81	0.82	0.86
CIFAR-100	0.96	0.75	0.69	0.73	0.68	0.86	0.83	0.85	0.82
ImageNet16-120	0.93	0.65	0.84	0.87	0.76	0.78	0.88	0.90	0.85

Fig. 2: Feature importance of the random forest prediction model trained on 80% of the data provided in [13] with all zero-cost proxies as features and multi targets being clean test accuracy and different adversarial accuracies on **CIFAR-10**.

5.2   Feature Importance

In the following we want to analyze the ability of ZCPs to be used as features to predict the robust accuracy as a single objective target, as well as the multi-objective target, for both clean and robust accuracy.

For that we split the architecture dataset (6 466 architectures from [13]) into 80% training and 20% test data to train a random forest regression model, that

takes as input a feature vector of all concatenated ZCPs. We use the default regression parameters, with 100 numbers of trees and the mean squared error as the target criterion. For a better overview, we will consider the attack ϵ value of $1/255$ for all the following experiments. We first analyze the prediction ability itself in Tab. 1 by means of the R^2 score of the prediction model on the test dataset. As we can see, the random forest prediction model is able to predict the single accuracy (clean and robust) and the multi-objective accuracies in a proper way, while the prediction of the clean accuracy seems to be the easiest task, where R^2 values of 0.93 to 0.97 are reached.

Next, we are interested in the feature importance of individual ZCPs. So far the ZCPs in NAS-Bench-Suite-Zero are motivated by their correlation with the resulting clean accuracy of the architectures. Yet, how well can they be transferred to the more challenging task of robust accuracy and even more challenging than that, the multi-objective task? High clean accuracy does not necessarily mean that a network is also robust. Therefore we use all ZCPs from NAS-Bench-Suite-Zero [14] and the robustness dataset [13], which we already presented in Sec. 3 as feature inputs for the random forest prediction model and calculate their importance as the mean variance reduction (Gini importance). Note that in this measure, the comparison of correlated features needs to be considered with caution. Fig. 2 visualizes the feature importance on the CIFAR-10 image datasets for all adversarial attacks with $\epsilon = 1/255$ for all three use cases: clean test accuracy, robust test accuracy, both test and robust accuracy. The latter one is used as the bar plot alignment. As we can see, `jacov` is the most important feature for both the clean test accuracy and the multi-objective task, except for FGSM. Interestingly, the Top 5 most important ZCPs are all different for all considered adversarial attacks.

If we look at the feature importance for CIFAR-100 (Fig. 3), we can see that the Top 4 most important features for clean and robust accuracy are always the same, with the Jacobian-based ZCP `nwot` being the most important one. Note here, for CIFAR-100 the rank correlation decreases the most for more difficult attacks.

When we turn now to ImageNet16-120, the ranking of the feature importance is the same for all considered attacks. Also there, a Jacobian-based ZCP, in this case `jacov`, is the most important feature.

We can see, that the most important features for clean accuracy are also important for the architecture's robust accuracy. In total, Jacobian-based ZCPs, especially `jacov` and `nwot`, seem to be the considerable measures for all different prediction tasks, ranging from single clean and robust accuracy prediction to the multi-objective prediction task. Noticeable is the high feature importance of `snip` for the single robust accuracy prediction task for both CIFAR datasets.

Interestingly, we can see that the task of robust accuracy prediction becomes more difficult on all three image datasets and attacks (see smaller R^2 values in Tab. 1) in comparison to the multi-objective task. In addition, Fig. 2, 3, and 4 give the impression that solely the most important feature could be used to predict the clean accuracy.
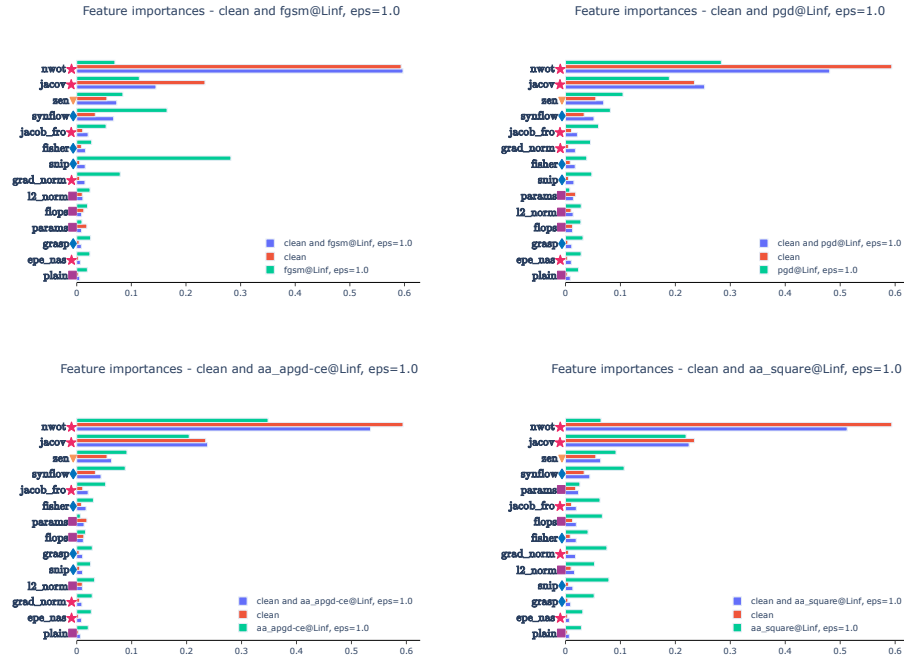
Fig. 3: Feature importance of the random forest prediction model trained on 80% of the data provided in [13] with all zero-cost proxies as features and multi targets being clean test accuracy and different adversarial accuracies on **CIFAR-100**.

Table 2: Test R^2 of the random forest prediction model using the **most important feature** for both single objectives and multi objectives on the clean test accuracy and the robust test accuracy for $\epsilon = 1/255$.

Dataset	Test Accuracy $\epsilon = 1/255$								
		Single Objective				Multi Objective			
	Clean	FGSM	PGD	APGD	Squares	Clean-FGSM	Clean-PGD	Clean-APGD	Clean-Squares
CIFAR-10	jacov 0.84	snip 0.41	jacov -0.31	jacov 0.002	snip 0.16	snip 0.37	jacov 0.26	jacov 0.42	jacov 0.54
CIFAR-100	nwot 0.74	snip 0.12	nwot 0.09	nwot 0.28	jacov -0.004	nwot 0.42	nwot 0.42	nwot 0.51	nwot 0.31
ImageNet16-120	jacov 0.61	jacov -0.003	jacov 0.58	jacov 0.64	jacov 0.44	jacov 0.31	jacov 0.60	jacov 0.63	jacov 0.52

Therefore, we analyze in Tab. 2 the prediction ability using only the most important feature for all considered tasks, respectively. As we can see the reduced R^2 values show that the joint consideration of several proxies is necessary to predict a model's robustness, while the clean accuracy can be regressed from only one such feature. Concluding, considering only one ZCP for the more difficult task of robust accuracy prediction (single and multi-objective) hinders the prediction ability.
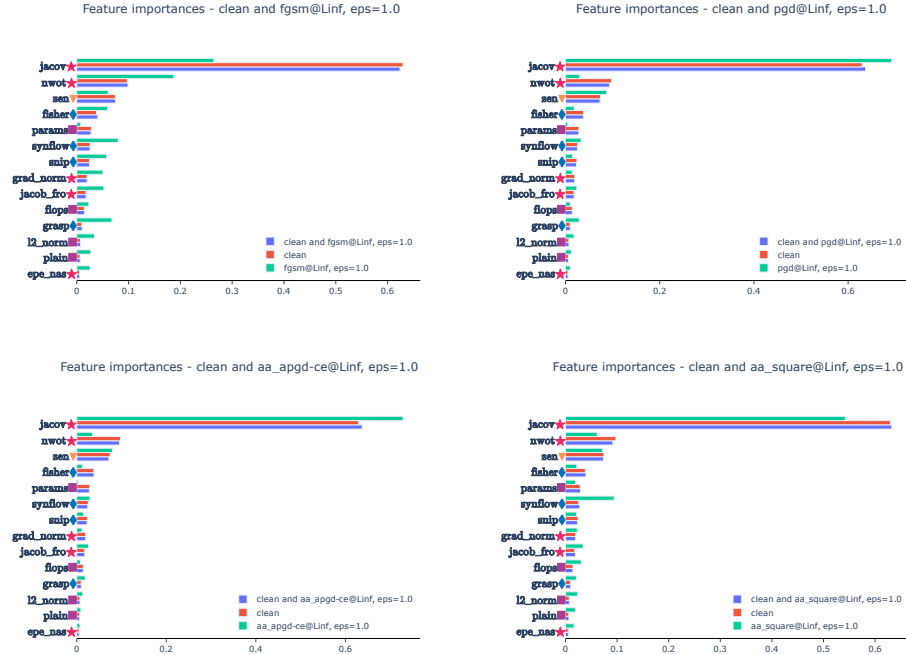
Fig. 4: Feature importance of the random forest prediction model trained on 80% of the data provided in [13] with all zero-cost proxies as features and multi targets being clean test accuracy and different adversarial accuracies on **ImageNet16-120**.

Table 3: Test R^2 of the random forest prediction model for the multi-objective task for $\epsilon = 1/255$ **without the most important feature** for the FGSM and PGD, respectively.

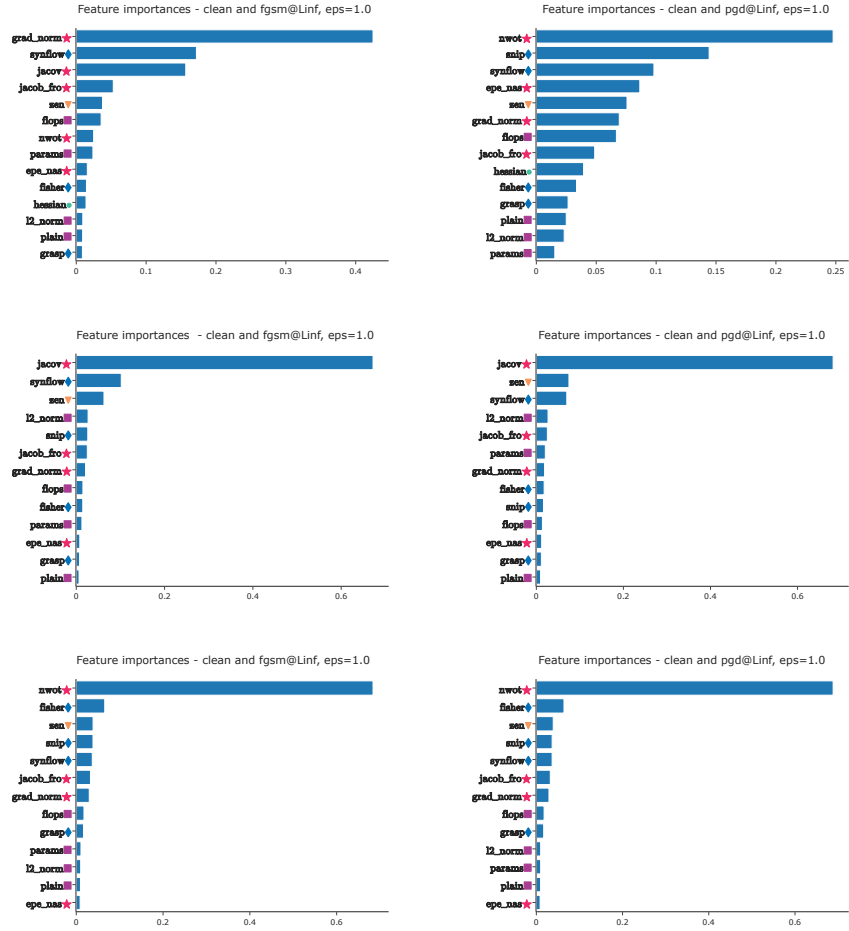| Dataset	Test Accuracy $\epsilon = 1/255$ Multi Objective	
	Clean-FGSM	Clean-PGD
CIFAR-10	0.92	0.79
CIFAR-100	0.87	0.83
ImageNet16-120	0.72	0.81

Fig. 5: Feature importance of the random forest prediction model trained on 80% of the data provided in [13] **without the most important feature** from Tables 2, 3, 4 for FGSM, and PGD at $\epsilon = 1/255$. (**top**): CIFAR-10, (**middle**): CIFAR-100, (**bottom**): ImageNet16-120.

5.3   Feature Importance excluding Top 1

From the previous section it became clear that using one zero cost proxy as a feature is not sufficient to predict the robustness of an architecture. This leads to the next question if the regression model can still make good predictions, using all ZCPs **except** the most important one.

The comparison of Tab. 1 with Tab.3 presents only a slight decrease in R^2 values. The prediction ability for the clean accuracy and FGSM robust accuracy on CIFAR-10, as well as for the clean accuracy and PGD accuracy on CIFAR-100 even remained the same.

We also plot the updated feature importances in Fig. 5 for all ZCPs without the most important one from the previous section for FGSM and PGD. CIFAR-100 (middle) and ImageNet16-120 (bottom) show a new most distinctive important feature, which is in both cases a Jacobian-based one. If we turn towards CIFAR-10 (top) there is no such distinctive important feature and rather a steady decrease in importance from one feature to the other.

6   Conclusion

In this paper, we presented an analysis of 15 zero-cost proxies with regard to their ability to act as performance prediction techniques for architecture robustness using a random forest. We made use of two provided datasets, NAS-Bench-Suite-Zero and a robustness dataset, which allows fast evaluations on the here-considered NAS-Bench-201 search space. Commonly, these zero-cost proxies are targeting the clean accuracy of architectures. Therefore, our analysis shows that the prediction of robustness is a more difficult task. Additionally, we investigated the feature importance of these zero-cost proxies. We found that, although the correlation of single zero-cost proxies with the robust accuracy of neural networks is quite weak, the signal contained in several such proxies considered simultaneously provides strong indication of a network's robustness. Thus, we conclude that (1) zero-cost proxies can, in principle, be used for NAS to search for robust architectures, and (2) unlike NAS for best clean accuracy, which usually employs a single proxy, NAS for robust architectures will need to leverage a multitude of zero-cost proxies in practice. At the same time, we see that the use of multiple proxies is beneficial even when regressing a network's clean accuracy. Future directions using zero-cost proxies in the NAS research should therefore consider the usage of several proxies with different underlying information instead of focusing only on one best proxy which should fit all tasks.

References

1. Abdelfattah, M.S., Mehrotra, A., Dudziak, L., Lane, N.D.: Zero-cost proxies for lightweight NAS. In: Proc. of the International Conference on Learning Representations (ICLR) (2021)
2. Chen, W., Gong, X., Wang, Z.: Neural architecture search on imagenet in four GPU hours: A theoretically inspired perspective. In: Proc. of the International Conference on Learning Representations (ICLR) (2021)
3. Chrabaszcz, P., Loshchilov, I., Hutter, F.: A downsampled variant of imagenet as an alternative to the CIFAR datasets. arXiv.org **abs/1707.08819** (2017)
4. Croce, F., Hein, M.: Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In: Proc. of the International Conference on Machine Learning (ICML) (2020)
5. Dong, M., Li, Y., Wang, Y., Xu, C.: Adversarially robust neural architectures. arXiv.org **abs/2009.00902** (2020)
6. Dong, X., Yang, Y.: Nas-bench-201: Extending the scope of reproducible neural architecture search. In: Proc. of the International Conference on Learning Representations (ICLR) (2020)
7. Elsken, T., Metzen, J.H., Hutter, F.: Neural architecture search: A survey. Journal of Machine Learning Research **20**, 55:1–55:21 (2019)
8. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: Proc. of the International Conference on Learning Representations (ICLR) (2015)
9. Guo, M., Yang, Y., Xu, R., Liu, Z., Lin, D.: When NAS meets robustness: In search of robust architectures against adversarial attacks. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
10. Hendrycks, D., Dietterich, T.: Benchmarking neural network robustness to common corruptions and perturbations. Proc. of the International Conference on Learning Representations (ICLR) (2019)
11. Hoffman, J., Roberts, D.A., Yaida, S.: Robust learning with jacobian regularization. arXiv.org **abs/1908.02729** (2019)
12. Hosseini, R., Yang, X., Xie, P.: DSRNA: differentiable search of robust neural architectures. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2021)
13. Jung, S., Lukasik, J., Keuper, M.: Neural architecture design and robustness: A dataset. In: Proc. of the International Conference on Learning Representations (ICLR) (2023)
14. Krishnakumar, A., White, C., Zela, A., Tu, R., Safari, M., Hutter, F.: Nas-bench-suite-zero: Accelerating research on zero cost proxies. In: Advances in Neural Information Processing Systems (NeurIPS) (2022)
15. Krizhevsky, A.: Learning multiple layers of features from tiny images. Tech. rep., University of Toronto (2009)
16. Kurakin, A., Goodfellow, I.J., Bengio, S.: Adversarial machine learning at scale. In: Proc. of the International Conference on Learning Representations (ICLR) (2017)
17. Lee, N., Ajanthan, T., Torr, P.H.S.: Snip: single-shot network pruning based on connection sensitivity. In: Proc. of the International Conference on Learning Representations (ICLR) (2019)
18. Li, Y., Vinyals, O., Dyer, C., Pascanu, R., Battaglia, P.W.: Learning deep generative models of graphs. arXiv.org **abs/1803.03324** (2018)

19. Lin, M., Wang, P., Sun, Z., Chen, H., Sun, X., Qian, Q., Li, H., Jin, R.: Zen-nas: A zero-shot NAS for high-performance image recognition. In: Proc. of the IEEE International Conference on Computer Vision (ICCV) (2021)
20. Liu, H., Simonyan, K., Yang, Y.: Darts: Differentiable architecture search. In: Proc. of the International Conference on Learning Representations (ICLR) (2019)
21. Lopes, V., Alirezazadeh, S., Alexandre, L.A.: EPE-NAS: efficient performance estimation without training for neural architecture search. In: International Conference on Artificial Neural Networks (ICANN) (2021)
22. Lukasik, J., Jung, S., Keuper, M.: Learning where to look - generative NAS is surprisingly efficient. In: Proc. of the European Conference on Computer Vision (ECCV) (2022)
23. Mellor, J., Turner, J., Storkey, A.J., Crowley, E.J.: Neural architecture search without training. In: Proc. of the International Conference on Machine Learning (ICML) (2021)
24. Mok, J., Na, B., Choe, H., Yoon, S.: Advrush: Searching for adversarially robust neural architectures. In: Proc. of the IEEE International Conference on Computer Vision (ICCV) (2021)
25. Ning, X., Tang, C., Li, W., Zhou, Z., Liang, S., Yang, H., Wang, Y.: Evaluating efficient performance estimators of neural architectures. In: Advances in Neural Information Processing Systems (NeurIPS) (2021)
26. Real, E., Aggarwal, A., Huang, Y., Le, Q.V.: Regularized evolution for image classifier architecture search. In: Proc. of the Conference of Artificial Intelligence (AAAI) (2019)
27. Ru, B., Wan, X., Dong, X., Osborne, M.: Interpretable neural architecture search via bayesian optimisation with weisfeiler-lehman kernels. In: Proc. of the International Conference on Learning Representations (ICLR) (2021)
28. Shen, Y., Li, Y., Zheng, J., Zhang, W., Yao, P., Li, J., Yang, S., Liu, J., Cui, B.: Proxybo: Accelerating neural architecture search via bayesian optimization with zero-cost proxies. arXiv.org **abs/2110.10423** (2021)
29. Tanaka, H., Kunin, D., Yamins, D.L.K., Ganguli, S.: Pruning neural networks without any data by iteratively conserving synaptic flow. In: Advances in Neural Information Processing Systems (NeurIPS) (2020)
30. Turner, J., Crowley, E.J., O'Boyle, M.F.P., Storkey, A.J., Gray, G.: Blockswap: Fisher-guided block substitution for network compression on a budget. In: Proc. of the International Conference on Learning Representations (ICLR) (2020)
31. Wang, C., Zhang, G., Grosse, R.B.: Picking winning tickets before training by preserving gradient flow. In: Proc. of the International Conference on Learning Representations (ICLR) (2020)
32. Wen, W., Liu, H., Chen, Y., Li, H.H., Bender, G., Kindermans, P.: Neural predictor for neural architecture search. In: Proc. of the European Conference on Computer Vision (ECCV) (2020)
33. White, C., Neiswanger, W., Savani, Y.: Bananas: Bayesian optimization with neural architectures for neural architecture search. In: Proc. of the Conference of Artificial Intelligence (AAAI) (2021)
34. White, C., Safari, M., Sukthanker, R., Ru, B., Elsken, T., Zela, A., Dey, D., Hutter, F.: Neural architecture search: Insights from 1000 papers. arXiv.org **abs/2301.08727** (2023)
35. White, C., Zela, A., Ru, B., Liu, Y., Hutter, F.: How powerful are performance predictors in neural architecture search? In: Advances in Neural Information Processing Systems (NeurIPS) (2021)

36. Wu, J., Dai, X., Chen, D., Chen, Y., Liu, M., Yu, Y., Wang, Z., Liu, Z., Chen, M., Yuan, L.: Stronger nas with weaker predictors. In: Advances in Neural Information Processing Systems (NeurIPS) (2021)
37. Xiang, L., Dudziak, L., Abdelfattah, M.S., Chau, T., Lane, N.D., Wen, H.: Zero-cost proxies meet differentiable architecture search. arXiv.org **abs/2106.06799** (2021)
38. Zhao, P., Chen, P., Das, P., Ramamurthy, K.N., Lin, X.: Bridging mode connectivity in loss landscapes and adversarial robustness. In: Proc. of the International Conference on Learning Representations (ICLR) (2020)
39. Zoph, B., Le, Q.V.: Neural architecture search with reinforcement learning. In: Proc. of the International Conference on Learning Representations (ICLR) (2017)