Efficient Activation Function Optimization through Surrogate Modeling

Garrett Bingham and Risto Miikkulainen

Google DeepMind · The University of Texas at Austin · cognizant

Abstract

This paper improves the state of the art in activation function design through three steps:

1. The benchmark datasets `Act-Bench-CNN`, `Act-Bench-ResNet`, and `Act-Bench-ViT` were created by training convolutional, residual, and vision transformer architectures from scratch with 2,913 activation functions.
2. A surrogate model was created to predict activation function performance based on the spectrum of the Fisher information matrix associated with the model's predictive distribution at initialization and the activation function's output distribution.
3. The surrogate was used to discover improved activation functions in several real-world tasks.

The approach (called AQuaSurF) produced a surprising finding: a sigmoidal design that outperformed all other activation functions. This discovery challenges the status quo of always using rectifier nonlinearities in deep learning.

Activation Function Benchmark Datasets

- Each dataset contains training results for 2,913 unique activation functions when paired with different architectures and tasks: All-CNN-C on CIFAR-10, ResNet-56 on CIFAR-10, and MobileViTv2-0.5 on Imagenette.

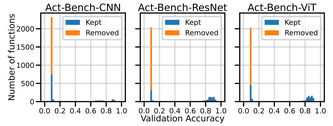


Figure 1: Distribution of validation accuracies with 2,913 unique activation functions from the three benchmark datasets. Many activation functions result in failed training (indicated by the chance accuracy of 0.1), suggesting that searching for activation functions is a challenging problem. However, most of these functions have invalid FIM eigenvalues, and can thus be filtered out effectively.

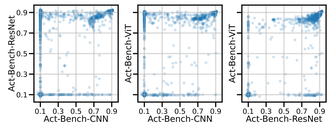


Figure 2: Distribution of validation accuracies across the benchmark datasets. Each point represents a unique activation function's performance on two of the three datasets. Some functions perform well on all tasks, while others are specialized.

Features and Distance Metrics

- Experimental data analysis on the benchmark datasets revealed two features that are highly predictive of activation function performance.
- Metrics are developed to allow for computing distances between activation functions in feature space.
- The Fisher information matrix (FIM) is defined as

$$\mathbf{F} = \mathop{\mathbb{E}}_{\substack{\mathbf{x}\sim\mathcal{D}_X \\ \mathbf{y}\sim P_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x},\boldsymbol{\theta})}} \left[\nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{y}, f(\mathbf{x};\boldsymbol{\theta})) \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{y}, f(\mathbf{x};\boldsymbol{\theta}))^\top \right]. \quad (1)$$

- Distances between two eigenvalue distributions are computed as a weighted layer-wise sum of 1-Wasserstein distances

$$d(f_\phi, f_\psi) = \textstyle\sum_{l=1}^{L} W_1(\mu_l, \nu_l)/w_l. \quad (2)$$

- The shape of an activation function ψ can be characterized by a vector of sample values $\psi(x)$, where $x \sim \mathcal{N}(0,1)$.
- Euclidean distance is used to compute the distance between two vectors of activation function outputs.

$$d(f_\phi, f_\psi) = \sqrt{\textstyle\sum_{i=1}^{n} (\phi(x_i) - \psi(x_i))^2 / n}, \quad x \sim \mathcal{N}(0,1). \quad (3)$$

Using the Features as a Surrogate

- The UMAP dimensionality reduction algorithm is used to map activation functions to low-dimensional embedding spaces based on their FIM eigenvalues and outputs.

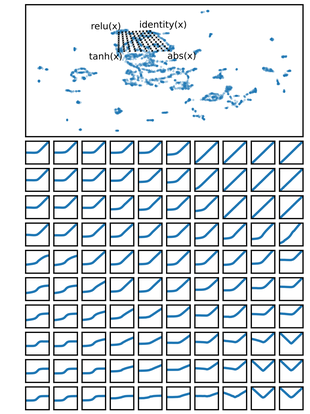


Figure 3: UMAP embedding of the 2,913 activation functions in the benchmark datasets. Each point stands for a unique activation function, represented by an 80-dimensional output feature vector. The embedding locations of four common activation functions are labeled. The black x's mark coordinates interpolating between these four functions, and the grid of plots on the bottom shows reconstructed activation functions at each of these points. UMAP interpolates smoothly between different kinds of functions, suggesting that it is a good approach for learning low-dimensional representations of activation functions.

- Using both FIM eigenvalues and function outputs together provides a better low-dimensional representation than using either feature alone.

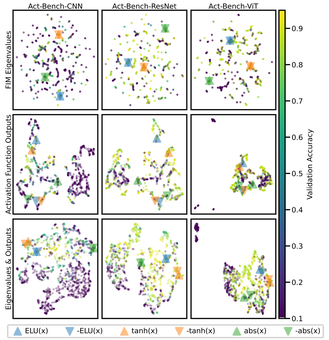


Figure 4: UMAP embeddings of activation functions for each dataset (column) and feature type (row). Each point represents a unique activation function; the points are colored by validation accuracy on the given dataset. The colored triangles identify the locations of six well-known activation functions. The areas of similar performance are more continuous in the bottom row; that is, using both FIM eigenvalues and activation function outputs provides a better low-dimensional representation than either feature alone.

- FIM eigenvalues help to identify activation functions that have different shapes but behave identically and result in the same performance, thus accelerating the search process.

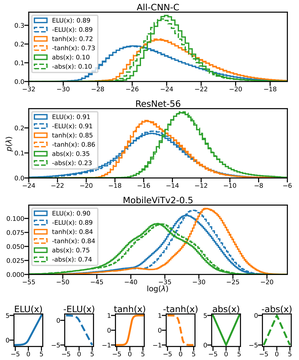


Figure 5: FIM eigenvalue distributions for different architectures and activation functions. The legends show the activation function and the corresponding validation accuracy in different tasks. Although negating an activation function changes its shape, it does not substantially change its behavior nor its performance. FIM eigenvalues capture this relationship between activation functions. The eigenvalues are thus useful for finding activation functions that appear different but in fact behave similarly, and these discoveries in turn improve the efficiency of activation function search.

Searching on the Benchmarks

- The benchmark datasets make it possible to experiment with different search algorithms and conduct repeated trials to understand the statistical significance of the results.

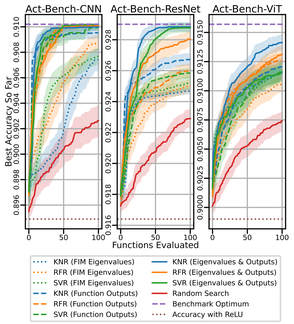


Figure 6: Search results on the three benchmark datasets. Each curve represents a different search algorithm (KNR, RFR, or SVR) utilizing a different UMAP feature (FIM eigenvalues, function outputs, or both; these features are visualized in Figure 4). The curves represent the validation accuracy of the best activation function discovered so far, averaged across 100 independent trials, and the shaded areas show the 95% confidence interval around the mean. In all cases, regression with UMAP features outperforms random search, and searching with both eigenvalues and outputs outperforms searching with either feature alone. Of the three regression algorithms, KNR performs the best, rapidly surpassing ReLU and quickly discovering near-optimal activation functions in all benchmark tasks. Thus, the features make it possible to find good activation functions efficiently and reliably even with off-the-shelf search methods.

Search Space Visualizations

- UMAP projections reveal the structure of the search space.

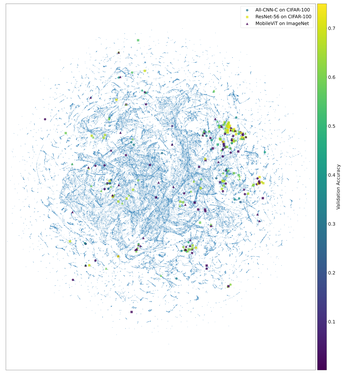


Figure 7: Low-dimensional UMAP representation of the 425,896 function search space. The activation functions are embedded according to their outputs; each point represents a unique function. The larger points represent activation functions that were evaluated during the searches; they are colored according to their validation accuracy. Although the space is vast, the searches require only tens of evaluations to discover good activation functions.

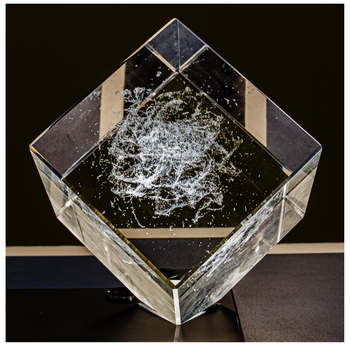


Figure 8: A photograph of a three-dimensional scatter plot laser-engraved into a physical crystal cube. Each point represents one of the unique 425,896 unique activation functions in the search space. Points are arranged according to a 3D UMAP projection according to activation function outputs; the points are the same as those shown in Figure 7. The cube shows the size and complexity of the search space, and the 1D and 2D manifolds reveal the underlying structure.

Scaling Up the Datasets and Search Space

- In this real-world task, activation functions were discovered for All-CNN-C on CIFAR-100, ResNet-56 on CIFAR-100, and MobileViTv2-0.5 on ImageNet.
- A larger search space with 425,896 unique activation functions was searched.

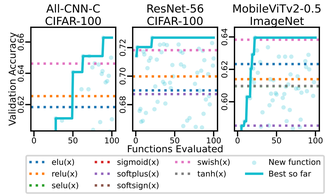


Figure 9: Progress of activation function searches. Each point represents the validation accuracy with a unique activation function, and the solid line indicates the performance of the best activation function found so far. AQuaSurF discovers new activation functions that outperform all baseline functions in every case.

Table 1: Accuracy with different activation functions. The CIFAR-100 results show the median test accuracy from three runs, and the ImageNet results show the validation accuracy from a single run. AQuaSurF discovers novel activation functions that outperform all baselines in every case. This result demonstrates both that good functions matter, and the power of optimizing them to the task.

All-CNN-C on CIFAR-100	
HardSigmoid(HardSigmoid(x)) · ELU(x)	**0.6990**
σ(Softsign(x)) · ELU(x)	0.6950
Swish(x)/SELU(1)	0.6931
ELU	0.6312
ReLU	0.6897
SELU	0.0100
sigmoid	0.0100
Softplus	0.6563
Softsign	0.2570
Swish	0.6913
tanh	0.3757

ResNet-56 on CIFAR-100	
Swish($-2x$)	**0.7469**
SELU(sinh($e^{\arctan(x)} - 1$))	0.7458
$x \cdot$ erfc(ELU(x))	0.7419
ELU	0.7411
ReLU	0.7348
SELU	0.6967
sigmoid	0.5766
Softplus	0.7397
Softsign	0.6624
Swish	0.7401
tanh	0.6754

MobileViTv2-0.5 on ImageNet	
$-x \cdot \sigma(x) \cdot$ HardSigmoid(x)	**0.6396**
ELU(Swish($-x$))	0.6394
Swish(x) · erfc(bessel_i0(x))	0.6336
ELU	0.6233
ReLU	0.6139
SELU	0.6096
sigmoid	0.5032
Softplus	0.5853
Softsign	0.5710
Swish	0.6383
tanh	0.6098

Transferring to a New Task

- The best activation functions from the previous experiment were transferred to a new task: ResNet-50 on ImageNet.

Table 2: ResNet-50 top-1 accuracy on ImageNet. Results are the median of three runs. The best activation functions discovered in the searches (Table 1) successfully transfer to this new task, with eight of the nine functions outperforming ReLU.

$-x \cdot \sigma(x) \cdot$ HardSigmoid(x)	**0.7776**
Swish(x)/SELU(1)	0.7771
Swish(x) · erfc(bessel_i0(x))	0.7755
σ(Softsign(x)) · ELU(x)	0.7734
SELU(sinh($e^{\arctan(x)} - 1$))	0.7719
HardSigmoid(HardSigmoid(x)) · ELU(x)	0.7718
ELU(Swish($-x$))	0.7679
Swish($-2x$)	0.7664
$x \cdot$ erfc(ELU(x))	0.7635
ReLU(x)	0.7660

New Architectures and Baseline Functions

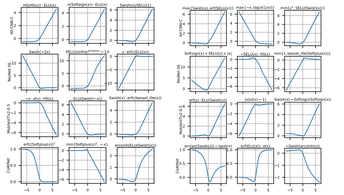
- The hybrid convolution and attention architecture CoAtNet provides a unique challenges for AQuaSurF.
- The activation functions ELiSH, GELU, HardSigmoid, Leaky ReLU, and Mish were added to the set of baseline functions and to the set of `unary` operators, forming a new search space to explore.

Table 3: CoAtNet validation accuracy on Imagenette. AQuaSurF finds novel functions that outperform all baselines.

erfc(Softplus(x))^2	**0.8907**
min{Softplus(x)^2, $-x$}	0.8861
arcsinh(ELU(Swish(x)))	0.8828
ELiSH	0.1000
ELU	0.8629
GELU	0.8841
HardSigmoid	0.8487
Leaky ReLU	0.8815
Mish	0.8762
ReLU	0.8772
SELU	0.8194
sigmoid	0.8586
Softplus	0.8678
Softsign	0.8530
Swish	0.8736
tanh	0.8415

Understanding the Discoveries

- Visually, many the best functions are similar to existing functions like ELU and Swish, with subtle changes in their saturation value, the slope of the positive segment, and the width and depth of the negative bump.
- However, some of the best discovered activation functions, including the top function for the CoAtNet experiment, employ properties uncommon among the usual deep learning activation functions: Some of them have discontinuous derivatives at $x = 0$; some do not saturate, but diverge as $x \to \pm\infty$; some of them contain positive bumps (in contrast to e.g. Swish, which features a negative bump).



(a) (b)

Figure 10: Sample activation functions discovered with AQuaSurF in the four searches. "HS" stands for HardSigmoid. (a) The top three functions (columns) discovered in each search (rows). Many of these functions are refined versions of existing activation functions like ELU and Swish. (b) Selected novel activation functions. All of these functions outperformed ReLU and are distinct from existing activation functions. Such designs may serve as a foundation for further improvement and specialization in new settings.

Discovering a Hybrid Rectifier-Sigmoidal Activation Function

- Rectifier activation functions typically outperform sigmoidal ones in modern tasks.
- Surprisingly, the best function discovered in the CoAtNet experiment, erfc(Softplus(x))^2, is sigmoidal in shape.
- Activation function input distributions reveal that the network uses the activation function like a rectifier at initialization and like a sigmoidal function after training.

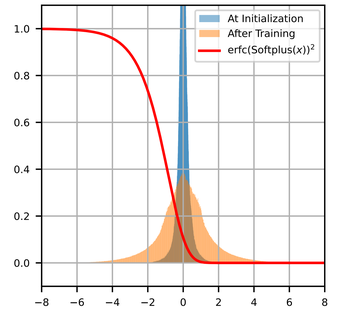


Figure 11: The best discovered function in the CoAtNet experiment, erfc(Softplus(x))^2, and its utilization by the network. The red curve shows the activation function itself, and the two histograms show the distributions of inputs to the activation function at initialization and after training, aggregated across all instances of the activation function in the entire network. The network uses the function like a rectifier at initialization and like a sigmoidal activation function after training. This result suggests that sigmoidal designs may be powerful after all, thus challenging the conventional wisdom.

Code and Benchmark Datasets

AQuaSurF search algorithm:
`https://github.com/cognizant-ai-labs/aquasurf`

Activation function benchmark datasets:
`https://github.com/cognizant-ai-labs/act-bench`

Contact

My website has links to my email, LinkedIn, Google Scholar, and CV.

garrettbingham.com