Squared Swish

Bojian Huang
1825636423@qq.com

Abstract

Swish(Prajit Ramachandran et al.,2017) is a great discovery that effectively alleviates gradient vanishing. Compared to ReLU(Xavier Glorot et al.,2011) piecewise linear structure, Swish curve is more continuous and can better fit complex function mappings, especially when dealing with high-dimensional nonlinear problems. Compared to ReLU "hard truncation" (negative input is directly set to 0), Swish's suppression of negative signals is softer, reducing information loss. However, under the same training configuration, Swish is convergence stability is slightly lower than ReLU. Since Swish performs so well, is there any variant of Swish that performs better than it. So I began to explore. After extensive experimental testing, I discovered Swish's variant Squared Swish, whose formula It is f (x)=x * sigmoid (x) * sigmoid (x), which performs better than Swish in some models.

1. Introduction

What I need to do is to use a large number of model tests to compare ReLU, Swish, and S-Swish, in order to verify whether S-Swish performs well.
For example, the popular small network MobileNetV3_Small (Andrew Howard et al.,2019),DenseNet100_12(Gao Huang et al.,2018),MobileNetV2(Mark Sandler et al.,2019),GhostNet(Kai Han et al.,2020), Add a new test dataset CIFAR100 (Alex Krizhevsky et al., 2009), and compare s-Swish with commonly used algorithms on the above test set.

2. Activation Function Compare

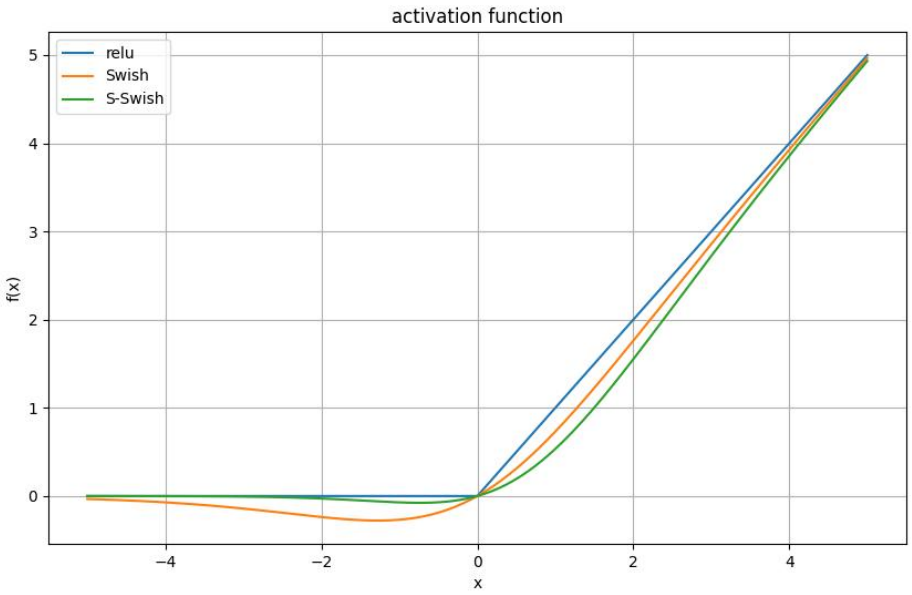

Figure 1：compare three activation function

The ReLU formula is f (x)=max (0, x). According to Figure 1, when x ≥ 0, the output is equal to the input (linearly increasing), and when x<0, the output is fixed at 0. The blue line in the figure reflects the characteristics of "flat negative half axis and straight positive half axis", which is easy to calculate and can alleviate gradient vanishing, but there is a problem of "ReLU death" (negative input causing neurons to not activate). The Swish formula is generally f (x)=x * σ (x) (where σ (x) is a Sigmoid function), which is a smooth nonlinear function. The orange line in the figure shows small fluctuations on the negative axis (due to Sigmoid characteristics), while the positive axis increases approximately linearly, making it more flexible than ReLU and often exhibiting better fitting ability in experiments. The S-Swish formula is f (x)=x * σ (x) * σ (x). The overall trend of the green line is similar to Swish, but there are slight differences in shape (such as the slope of the negative and low positive half axis curves).

3. Activation Function Derivative Compare

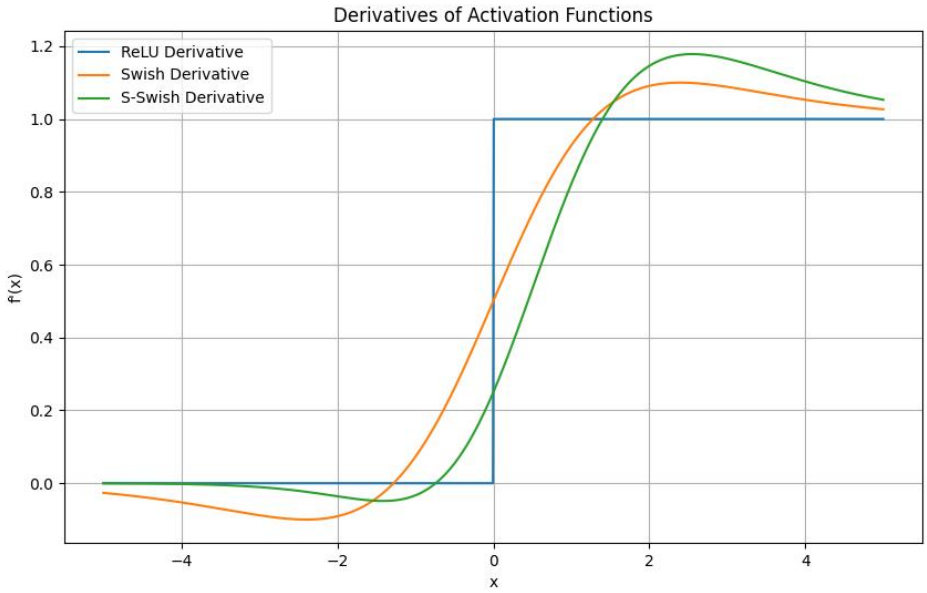

Figure 2：compare three activation function derivatives

The ReLU function expression is f (x)=max (0, x). According to Figure 2, its derivative is 1 when x>0 and 0 when x<=0. From the graph, it can be seen that when x>0, the derivative remains stable at 1, and when x<=0, it is 0, in the form of piecewise constants. The derivative is simple and stable, computationally efficient, with a constant positive interval derivative of 1, which can alleviate gradient vanishing (compared to sigmoid, etc.); But if the negative interval derivative is 0, it will cause some neurons to "die" (the gradient cannot be updated when it is 0 during training). The Swish function typically takes the form f (x)=x * Sigmoid (βx) (βoften takes 1), and its derivative is derived in a composite form combining sigmoid and its derivative. As can be seen in the figure, the derivative curve is continuously smooth. When x<0, it first becomes negative and fluctuates, gradually rising, and then increases to the peak when x>0, and then slowly decreases to reach a stable value. Due to the continuous derivative, gradient propagation during training is smoother, which to some extent avoids the "neuron death" problem of ReLU and helps improve the model's expressive ability; However, the computational complexity is slightly higher than ReLU due to the involvement of sigmoid function calculations. S-Swish is a variant of Swish, and the derivative curve is also continuous. From the graph, the overall trend is similar to Swish, but there are differences in details such as the rate of change and

peak height, such as the slope of the rising phase and the position of the peak height, indicating that its derivative characteristics have been adjusted to meet different model requirements. Inheriting the advantages of continuous derivatives and optimizing gradient flow in certain scenarios (such as specific data distributions and model structures) through adjustments; However, the effect gain brought by variants needs to be verified in conjunction with specific tasks, and may also increase the cost of hyperparameter debugging.

4. Activation Function Second Derivative Compare

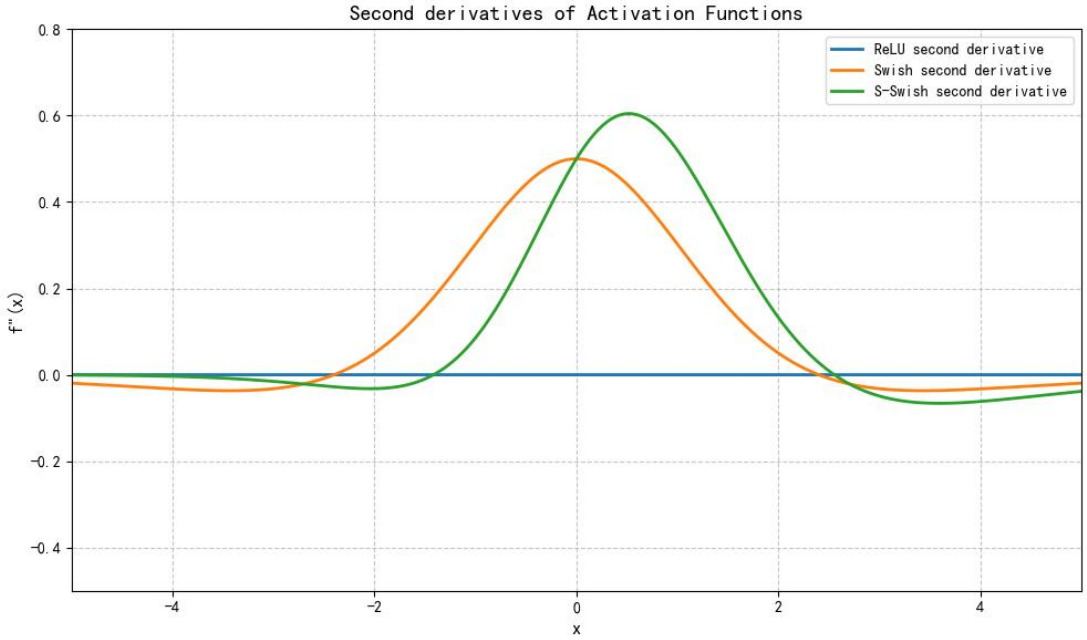

Figure 3： compare three activation function second derivatives

The second derivative f’’(x) reflects the bending direction of the function graph by describing the rate of change of the first derivative.

When f’(x)>0, the function graph is convex upward at that point, with a shape similar to a "U" shape. At this point, the tangent slope of the function increases with the increase of x.

When f’(x)<0, the function graph is convex downward at that point, with a shape similar to a "∩" shape. At this point, the tangent slope of the function decreases as x increases.

When f’(x)=0 and the sign of the second derivative on both sides of the point changes, the point is called the inflection point, which is the critical point where the concavity and convexity of the function changes.

Near the interval x<-2: The green curve (S-Swish second derivative) is below 0, that is, f’(x)<0. The function is convex in this interval, and the image has a downward bending trend.

Interval -2<x<2: The green curve first rises to the positive region (f’(x)>0) and then decreases, indicating that the function first becomes convex (due to the positive second derivative, the curve bends upwards), and then may switch between convexity and concavity; The middle peak reflects the area with the highest degree of curvature, and the higher the peak, the steeper the local curvature.

Near the interval x>2: The green curve returns to below 0 (f’(x)<0), and the function exhibits a downward convex feature, causing the image to curve downwards.

Turning point identification: The point where the second derivative changes from positive to negative or negative to positive is the turning point. From the graph, the intersection point between the green curve and the x-axis (excluding the endpoint) is a possible turning point, corresponding to the position

where the concavity and convexity of the function switch.

Compared to ReLU (blue, with an approximate second derivative of 0, the function is mostly linear or "hard" nonlinear in most intervals, with a single concavity and convexity) and Swish (orange, with a different shape from S-Swish, reflecting different activation functions due to their second derivative characteristics, concavity and curvature patterns), S-Swish exhibits a complex concavity and convexity pattern of "first downward convex, then upward convex, and then downward convex" due to the positive and negative changes in the second derivative.

5. Experiment

Function	Small_mobilenetV3
S-Swish	**80.30**
H-Swish	78.20
ReLU	79.61
Swish	78.67

Table1 :CIFAR-10 accuracy in small_mobilenetV3

In Table 1, the results obtained from the validation of the model using four activation functions on the CIFAR-10 set after training Small_mobilenetV3 show that S-Swish performs the best, while H-Swish performs the worst.

Function	mobilenetv2
S-Swish	**90.05**
ReLU	87.93
Swish	89.90

Table2:CIFAR-10 accuracy in mobilenetv2

In Table 2, three activation functions were used to train the MobilenetV2 model on the CIFAR-10 set, and the results showed that S-Wish still performed the best.

Function	desnet
ReLU	**95.99**
Swish	95.66
S-Swish	95.89

Table3 :CIFAR-10 accuracy in desnet

In Table 3, three activation functions were used to train the desnet model on the CIFAR-10 set, and the results showed that ReLU still performed the best, followed by S-Wish.

Function	ghostnet
ReLU	87.56
Swish	**88.38**
S-Swish	88.06

Table4 :CIFAR-10 accuracy in ghostnet

In Table 4, three activation functions were used to train the model for ghostnet on the CIFAR-10 set, and the results were validated. It can be seen from the results that Swish first appeared as the top performer, followed by S-Wish, and ReLU performed the worst.

Function	Small_mobilenetV3
S-Swish	**54.89**
H-Swish	54.87
ReLU	54.86
Swish	54.01

Table5 :CIFAR-100 accuracy in small_mobilenetV3

In Table 5, three activation functions were used to validate the model trained on the CIFAR-100 set for Small_mobilenetV3. From the results, it can be seen that S-Swish once again exhibited anti overshoot and ranked first, but the difference was very small. Surprisingly, Swish ranked last.

Function	mobilenetV2
S-Swish	66.70
ReLU	65.13
Swish	**67.36**

Table6 :CIFAR-100 accuracy in mobilenetV2

In Table 6, three activation functions were used to train the model for mobilenetV2 on the CIFAR-100 set, and the results showed that Swish returned to the top position.

Function	ghostnet
S-Swish	61.44
ReLU	60.66
Swish	**61.71**

Table7 :CIFAR-100 accuracy in mobilenetV3

In Table 7, three activation functions were used to train the model for mobilenetV2 on the CIFAR-100 set, and the results showed that Swish still ranked first.

Based on the above seven experiments, we can draw the following conclusion from the results: S-Swish has a certain advantage in a relatively small number of classifications, but in a large number of classifications, S-Swish does not perform well, but it is still very close to Swish and better than ReLU. However, there is an exception, that is, in SmallhmodelnetV3, S-Swish's accuracy exceeds Swish.

6. Conclusion

From the above experiments, we can see that the performance of S-Swish has always been better than ReLU, and there are ups and downs in the comparison with Swish. Although the calculation process of S-Swish is more complex than Swish, and its performance will only be better under certain conditions, such as in the case of a small number of classifications, if Swish's performance is not very good, S-Swish may be a good choice. This study mainly compares the S-Swish structure with other activation functions in practical applications, and finally obtains preliminary conclusions. Under certain conditions, S-Swish may perform better than other activation functions such as Swish.

References

[1]Prajit Ramachandran, Barret Zoph, Quoc V. Le.SEARCHING FOR ACTIVATION FUNCTIONS,arXiv:1707.04873,2017.
[2]Xavier Glorot,Antoine Bordes,Yoshua Bengio. Deep Sparse Rectifier Neural Networks,2011.
[3]Andrew Howard,Mark Sandler,Grace Chu,Liang-Chieh Chen,Bo Chen,Mingxing Tan,Weijun Wang,Yukun Zhu,Ruoming Pang,Vijay Vasudevan,Quoc V. Le,Hartwig Adam.Searching for MobileNetV3,arXiv:1905.02244v5,2019.
[4]Gao Huang,Zhuang Liu,Laurens van der Maaten,Kilian Q. Weinberger.Densely Connected Convolutional Networks,arXiv:1608.06993v5,2018.
[5]Mark Sandler Andrew Howard Menglong Zhu Andrey Zhmoginov Liang-Chieh Chen.MobileNetV2: Inverted Residuals and Linear Bottlenecks,arXiv:1801.04381v4,2019.
[6]Kai Han1 Yunhe Wang1 Qi Tian1∗ Jianyuan Guo2 Chunjing Xu1 Chang Xu3.GhostNet: More Features from Cheap Operations,arXiv:1911.11907v2 [cs.CV],2020.
[7]Alex Krizhevsky.GhostNet: Learning Multiple Layers of Features from Tiny Images,arXiv:1911.11907v2 [cs.CV],2009.