



UNIVERSIDADE FEDERAL DA PARAÍBA - UFPB
CENTRO DE INFORMÁTICA - CI
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO



Kelvin Brenand - 20180005270

Relatório Sobre a Avaliação de Funções de Ativação Para Classificação de Achados em Imagens Radiográficas

João Pessoa

12 de dezembro de 2022

Kelvin Brenand - 20180005270

Relatório Sobre a Avaliação de Funções de Ativação Para Classificação de Achados em Imagens Radiográficas

Relatório referente à avaliação de funções de ativação para classificação de achados em imagens radiográficas, que corresponde ao projeto final da disciplina Processamento Digital de Imagens.

Prof. Dr. Leonardo Vidal Batista

Universidade Federal da Paraíba - UFPB
Centro de Informática - CI
Curso de Graduação em Ciência da Computação

João Pessoa

12 de dezembro de 2022

Sumário

1	Introdução	4
2	Fundamentação Teórica	4
2.1	Redes Neurais Covolucionais	4
2.2	Funções de Ativação	5
2.2.1	Sigmoid	6
2.2.2	Tanh	7
2.2.3	ReLU	7
2.2.4	Selu	8
2.2.5	Swish	9
3	Objetivos	9
4	Materiais e Métodos	10
5	Resultados	11
6	Discussão	11
7	Conclusões	12

Lista de Figuras

1	Etapas de processamento de uma imagem para entrada em uma CNN	5
2	Processamento de uma imagem por uma CNN	5
3	Função sigmoid	6
4	Função Tanh	7
5	Função ReLU	8
6	Função SELU	8
7	Função Swish	9
8	Exemplo de imagens do CheXpert	10
9	Detecção das patologias nas imagens.	12

1 Introdução

Reconhecimento de padrões é uma tarefa cumprida com facilidade por humanos, exigindo pouquíssimo esforço e não se limitando ao campo visual, mas detectando também padrões com outros sentidos. Tornar um computador capaz de fazer o mesmo é uma tarefa árdua que vem sendo amplamente desenvolvida nas últimas décadas. Fazer o computador reconhecer padrões requer o estudo de Redes Neurais. O projeto aqui descrito consiste do projeto final relativo à disciplina de Introdução ao Processamento Digital de Imagens, ministrada pelo professor Leonardo Vidal Batista, do curso ciência da computação da Universidade Federal da Paraíba (UFPB).

Tal projeto consiste em aplicar de forma prática os conhecimentos teóricos ofertados pela disciplina. Sendo isso feito através da avaliação do desempenho de 5 modelos de redes neurais aplicados sobre imagens radiográficas. A diferença entre os modelos está na função de ativação utilizada, sendo as funções de ativação em questão a RElu, Sigmoid, Swish, Tahn e a Selu.

2 Fundamentação Teórica

As Redes Neurais Artificiais, também chamadas de “Redes Neurais”, consistem em uma estrutura de nós (ou neurônios artificiais) em camadas e interligados. A informação de entrada do sistema alimenta a camada de entrada, e a saída desta alimenta a camada seguinte, que alimenta a camada seguinte, e assim sucessivamente, até a camada de saída.

Os neurônios fazem uma transformação linear na entrada pelos pesos e bias. O peso multiplica uma informação computada da entrada, indicando a sua influência na saída da unidade. O bias aumenta ou diminui o valor da entrada, de forma a transladar a função de ativação no eixo. A transformação não linear é feita pela função de ativação. Esta passagem da informação entre as camadas até a camada de saída é conhecida como propagação direta. Se o resultado gerado estiver longe do valor esperado, os pesos e bias dos neurônios são atualizados com base no erro. Este processo é conhecido como *backpropagation*. Uma vez que todos os dados passaram por este processo, os pesos e bias finais são usados para previsões.

2.1 Redes Neurais Covolucionais

As redes neurais convolucionais (CNN) são muito semelhantes às redes neurais comuns. As arquiteturas das CNNs assumem explicitamente que as entradas são imagens, o que nos permite

codificar certas propriedades na arquitetura. Isso torna a função de encaminhamento mais eficiente para implementar e reduz consideravelmente a quantidade de parâmetros na rede.

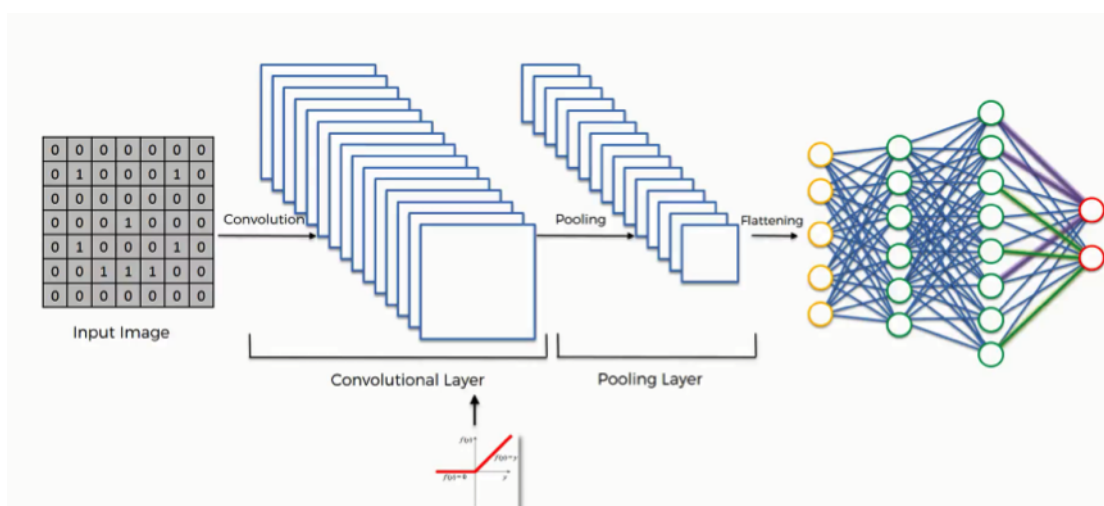


Figura 1: Etapas de processamento de uma imagem para entrada em uma CNN

Como descrevemos acima, uma CNN simples é uma sequência de camadas, e cada camada de uma CNN transforma um volume de ativações em outro por meio de uma função diferenciável. Usamos três tipos principais de camadas para construir arquiteturas CNN: camada convolucional, camada de *pooling* e camada totalmente conectada (exatamente como visto em redes neurais comuns).

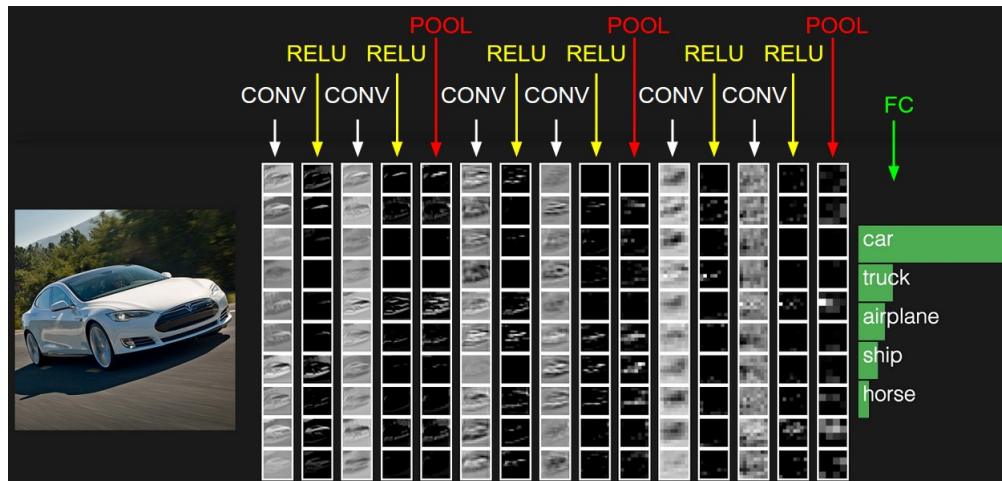


Figura 2: Processamento de uma imagem por uma CNN

2.2 Funções de Ativação

Um dos componentes mais importantes do neurônio artificial é a função de ativação ou transferência. As transformações lineares, feitas com o peso e o bias, são limitadas para problemas complexos e realizam grandes alterações na entrada. As funções de ativação, que realizam transformações não lineares, permitem que pequenas mudanças nos pesos e bias causem apenas uma pequena alteração

no *output*. Esse é o fato crucial que permitirá que uma rede de neurônios artificiais aprenda, se ajustando adequadamente. Uma função de ativação é dada por:

$$y = \sigma\left(\sum_i W_i * x_i\right) + b \quad (1)$$

onde σ é a função de ativação, W são os pesos, x os inputs e b o viés.

As funções de ativação limitam a amplitude válida do sinal de saída do neurônio e, normalmente, esta amplitude normalizada se encontra em um intervalo $[0, 1]$ ou, em alguns casos, $[-1, 1]$ e, assim, basicamente decidem se um neurônio deve ser ativado ou não. Ou seja, se a informação que o neurônio está recebendo é relevante para a informação fornecida ou deve ser ignorada. A seguir, serão apresentados os 5 tipos de funções de ativação utilizadas.

2.2.1 Sigmoid

A função Sigmóide é dada por:

$$f(x) = \frac{1}{1 + e^{-x}} \quad (2)$$

A função varia de 0 a 1 tendo um formato S. Ela essencialmente tenta empurrar os valores de Y para os extremos. Esta é uma qualidade muito desejável quando tentamos classificar os valores para uma classe específica. A função sigmoid ainda é amplamente utilizada até hoje, mas temos problemas quando os gradientes se tornam muito pequenos. Isso significa que o gradiente está se aproximando de zero e a rede não está realmente aprendendo.

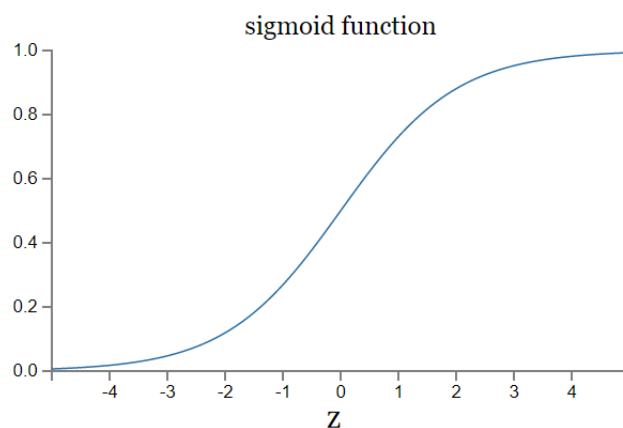


Figura 3: Função sigmoid

2.2.2 Tanh

A função Tanh (ou *Hiperbolic Tanget Function*) é dada por:

$$f(x) = \frac{2}{(1 + e^{-2x}) - 1} \quad (3)$$

A função de ativação Tanh é uma função sigmoide tangente hiperbólica que tem um intervalo de -1 a 1. Ela é frequentemente usada em modelos de aprendizado profundo por sua capacidade de modelar limites não lineares. A função tanh é popular por sua simplicidade e pelo fato de não saturar para pequenas entradas como o sigmoid, o que significa que pode ser aplicada em diferentes escalas sem perder sua eficácia.

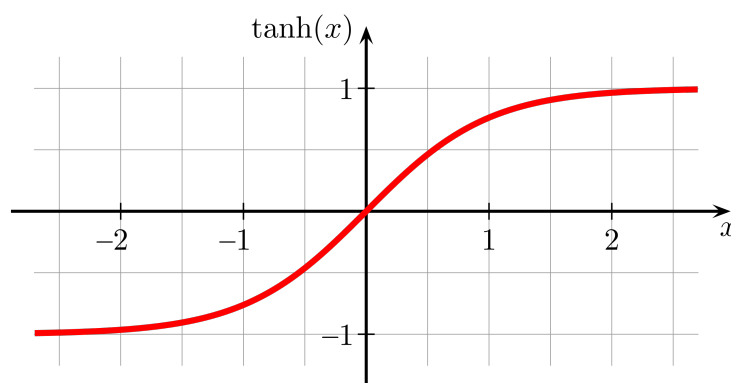


Figura 4: Função Tanh

2.2.3 ReLU

A função ReLU (ou Rectified Linear Unit) é dada por:

$$f(x) = \max(x, 0) \quad (4)$$

Ela fornece uma transformação não linear bem simples. Dado um elemento, a função é definida como o máximo desse elemento e 0. Ela retém apenas elementos positivos e descarta todos os elementos negativos definindo as ativações correspondentes para 0. Tornou-se a função de ativação padrão para muitos tipos de redes neurais porque um modelo que a utiliza é mais fácil de treinar e geralmente obtém melhor desempenho.

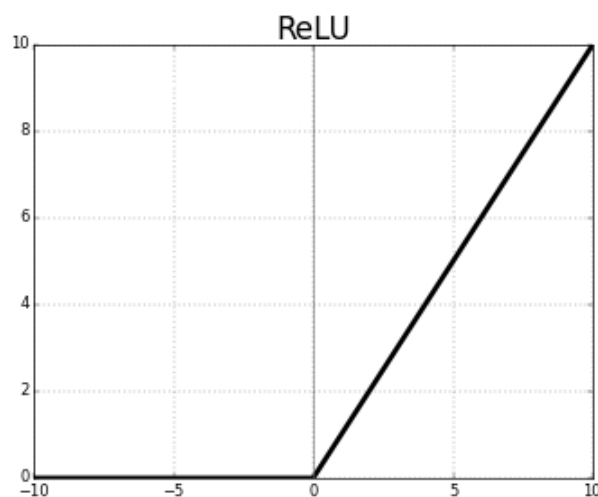


Figura 5: Função ReLU

2.2.4 Selu

A função SELU (ou Scaled Exponential Linear Units) é dada por:

$$f(x) = \lambda \begin{cases} x & \text{se } x > 0 \\ \alpha e^x - \alpha & \text{se } x \leq 0 \end{cases} \quad (5)$$

SELU é uma função de ativação de auto-normalização. É uma variante do ELU. A principal vantagem do SELU é que podemos ter certeza de que a saída será sempre padronizada devido ao seu comportamento de autonormalização. Isso significa que não há necessidade de incluir camadas de normalização em lote.

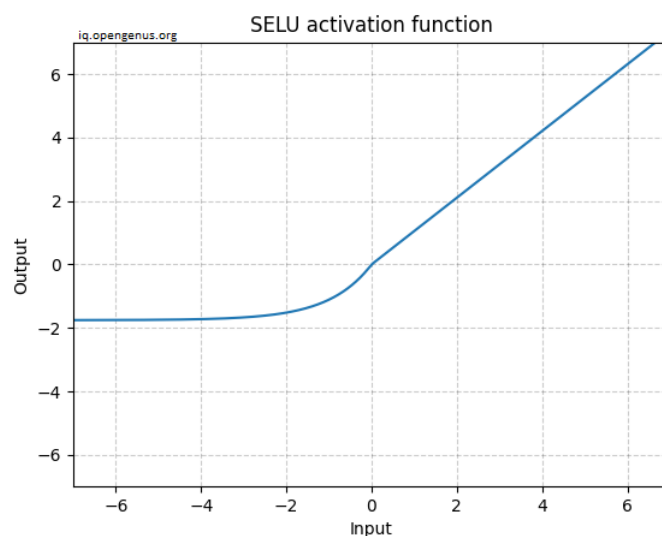


Figura 6: Função SELU

2.2.5 Swish

A função Swish é dada por:

$$f(x) = \frac{x}{1 + e^{-x}} \quad (6)$$

Swish é uma função de ativação suave, não muda repentinamente de direção como ReLU faz perto de x igual a zero. Em vez disso, ele se inclina suavemente de 0 para valores menores que 0 e depois para cima novamente. Swish é uma função que consistentemente se equipara ou supera ReLU em redes profundas aplicadas a uma variedade de problemas desafiadores, como classificação de imagens e tradução automática.

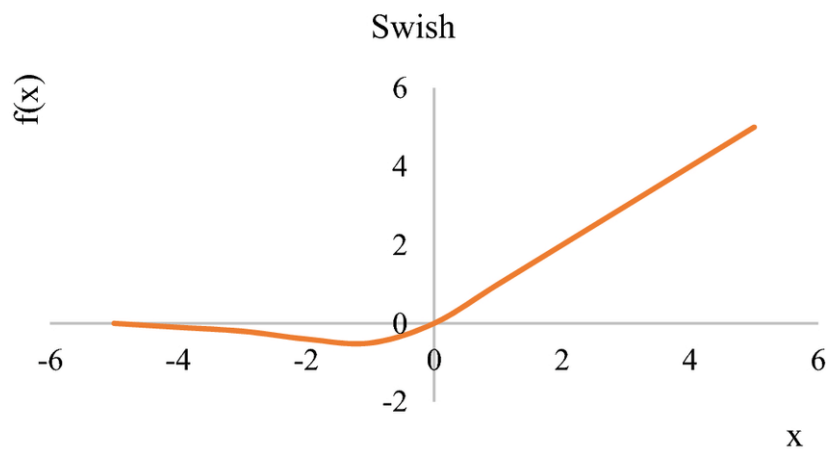


Figura 7: Função Swish

3 Objetivos

O objetivo buscado nesse trabalho é avaliar o desempenho de diferentes funções de ativação de modelos de redes neurais convolucionais. Para isso foi utilizada a métrica acurácia.

A acurácia é uma métrica para avaliar os modelos de classificação. Informalmente, a acurácia é a fração de previsões que nosso modelo acertou. Formalmente, acurácia tem a seguinte definição:

$$Acurcia = \frac{\text{Numero de predicoes corretas}}{\text{Numero total de predicoes}} \quad (7)$$

Para classificação binária, a acurácia também pode ser calculada em termos de positivos e negativos da seguinte forma:

$$Acurcia = \frac{TP + TN}{TP + TN + FP + FN} \quad (8)$$

Onde $TP = \text{True Positives}$, $TN = \text{True Negatives}$, $FP = \text{False Positives}$ e $FN = \text{False Negatives}$.

4 Materiais e Métodos

Os dados utilizados durante o treinamento e teste dos modelos foi o CheXpert. CheXpert é um grande conjunto de dados de radiografias de tórax e competição para interpretação automatizada de radiografia de tórax, que apresenta rótulos de incerteza e conjuntos de avaliação padrão de referência rotulados por radiologistas.

Esse *dataset* consiste em 224.316 radiografias de tórax de 65.240 pacientes, onde os exames radiográficos de tórax e os relatórios radiológicos associados foram coletados retrospectivamente no Stanford Hospital. Cada relatório foi rotulado para a presença de 14 observações como positivas, negativas ou incertas. Decidimos as 14 observações com base na prevalência nos relatórios e na relevância clínica.

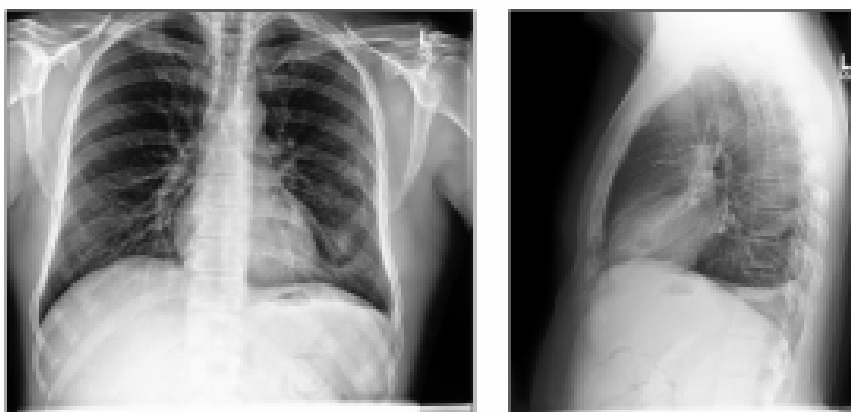


Figura 8: Exemplo de imagens do CheXpert

No entanto, devido à limitações de natureza técnica, o *dataset* precisou ser reduzido do que originalmente seria um problema de classificação multi-classe para um problema de classificação de apenas duas classes: Casos positivos ou casos negativos. Neste sentido, foram utilizados 3000 imagens de 400x400 pixels para o treinamento e 1300 imagens para o teste.

Para o desenvolvimento das funcionalidades em si foi utilizada a linguagem de programação Python em conjunto com as suas seguintes bibliotecas:

- PIL: para a extração dos dados dos pixels de imagens
- Numpy: para a vetorização e utilização de certas operações em matrizes
- TensorFlow: para aprendizado de máquina e inteligência artificial.

5 Resultados

Os resultado a seguir foi produzido utilizando uma classificação de duas classes, com 3000 imagens de 400x400 pixels para o treinamento e 1300 imagens para o teste. As imagens correspondem à exames radiográficos de tórax numa visão frontal.

Função de Ativação	Acurácia
Sigmoid	0.53
Tanh	0.55
ReLU	0.65
SELU	0.61
Swish	0.62

Tabela 1: Resultados

A estrutura da rede neural utilizada, onde variou-se apenas a função de ativação, foi a seguinte:

Layer (type)	Output Shape	Param #
Conv2D	(None, 199, 199, 32)	896
MaxPooling2D	(None, 99, 99, 32)	0
Conv2D	(None, 49, 49, 64)	18496
MaxPooling2D	(None, 24, 24, 64)	0
Flatten	(None, 36864)	0
Dense	(None, 64)	2359360
Dense	(None, 32)	2080
Dense	(None, 2)	66

Tabela 2: Arquitetura dos modelos de rede neural

Total params: 2,380,898, Trainable params: 2,380,898, Non-trainable params: 0.

6 Discussão

Através da análise da tabela 1, pode-se observar que o desempenho geral dos modelos foi relativamente baixo. Todos os modelos ficaram em volta dos 0.50-0.60 de acurácia, sendo o modelo com a função de ativação ReLU o melhor, com uma acurácia de 65%. Este resultado é esperado uma vez que esta função de ativação é o padrão da área, devido ao seu desempenho geral para CNNs. Funções de ativação mais simples como a Sigmoid acabaram tendo um desempenho inferior.

O baixo desempenho geral dos modelos pode ser explicado por alguns fatores do treinamento. O primeiro deles seria que, ao utilizar uma alta (mais que 3000) quantidade de imagens durante o treinamento e teste, o sistema operacional "crashava", impossibilitando o uso de um dos principais fatores que levam a um bom desempenho de CNNs: volume de dados. Também ligado ao primeiro

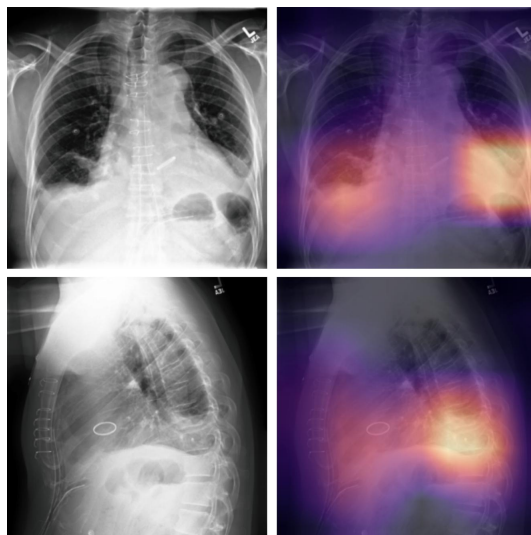


Figura 9: Detecção das patologias nas imagens.

fator, o segundo seria que ao tentar utilizar uma modelo de rede neural mais complexo, com o objetivo de melhorar o desempenho, o sistema novamente falhava. Dado o exposto, os resultados obtidos foram bons o suficiente para atingir o objetivo do projeto.

7 Conclusões

Este trabalho, desenvolvido como parte da disciplina Introdução ao Processamento Digital de Imagens, se propôs a documentar a avaliação do desempenho de diferentes funções de ativação de modelos de redes neurais convolucionais. A partir dos experimentos aqui documentados pôde-se observar o real e claro impacto das funções de ativação no desempenho dos modelos e, dessa maneira, pode-se atestar que o objetivo foi alcançado. Porém, ainda seria possível obter alguns aperfeiçoamentos para o código implementado, sendo o mais destacável a melhoria da performance através do uso de um computador mais potente, sendo este o maior gargalo do projeto.

Referências

Material de Aula - Leonardo Vidal Batista -

<https://sig-arq.ufpb.br/arquivos/202209712848df45247207774942b62d9/PDI20221.pdf> –

Acesso em: 10/12/2022.

Tensorflow - <https://www.tensorflow.org/> – Acesso em: 10/12/2022.