**RELATÓRIO DO LABORATÓRIO 08**

**Nome**: André Augusto Bernabé da Costa Marques **RA**: 22001640

1. **Descrição das Arquiteturas**

Este projeto apresenta a implementação de uma rede neural desenvolvida inteiramente sem o uso de bibliotecas específicas de IA, como TensorFlow ou PyTorch. O modelo é voltado para a classificação de imagens em bases de dados 28x28 pixels, sendo adaptável a diversos datasets, como Fashion MNIST e MNIST. Utiliza-se Python com as bibliotecas numpy para manipulação de arrays e matplotlib para visualização de dados.

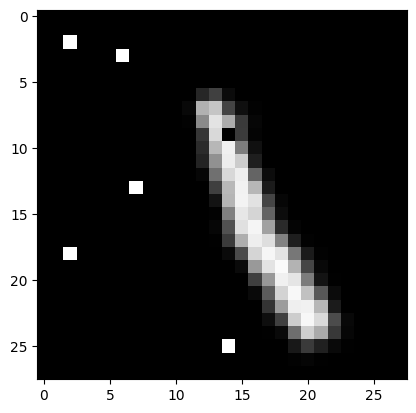
A rede neural implementada possui operações essenciais como propagação para frente, retropropagação e atualização de pesos e vieses, sendo possível simular o comportamento de uma rede com diferentes configurações de camadas. O modelo é flexível, adaptando-se a outros conjuntos de dados com facilidade.

O foco desse relatório é direcionado a uma técnica que possivelmente irá aumentar a confiabilidade e precisão da Rede Neural, o Data Augmentation. Seu objetivo é de modificar as imagens levemente, de modo que possa realizar um treino com mais casos de uso para a Rede Neural.

**Principais Funções**

Fora necessário criar uma função para o Data Augmentation, a função foi implementada por fora da Classe NeuralNetwork, porém poderia ter sido implementada dentro, se tornando um hiper-parâmetro a mais.

A função criada fora chamada de imageAugmentation(recebe como input uma imagem). Ela retorna a imagem inputada com uma pequena e aleatória rotação, translação e adição de ruído.



Após isso, o código passa por dois laços de For, passando todas as imagens retornadas modificadas em uma lista por uma quantidade x de vezes, podendo dessa forma, modificar as 60k imagens em quantas vezes e modificações quiserem. Após as listas estarem feitas, é feita uma concatenação da lista com o vetor original e então misturado com as labels corretas.

**Como o Modelo Funciona**

O modelo continua funcionando da mesma maneira como anteriormente, haja visto, que a única função adicionada é utilizada fora da Classe.

**Visualização dos Resultados**

A rede neural também oferece ferramentas para visualizar o desempenho do modelo:

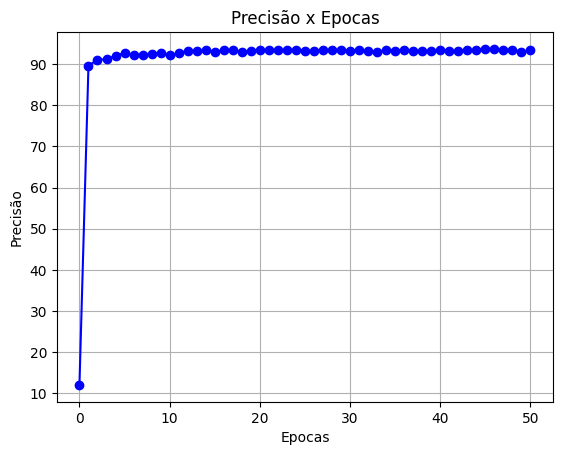
**Precisão por Épocas**: O gráfico "Precisão x Épocas" exibe a evolução da precisão do modelo ao longo das épocas de treinamento.

**Visualização dos Pesos:** O método verCamadaHidden permite visualizar os pesos da camada de entrada como imagens, fornecendo uma visão interessante de como os pesos se ajustam durante o treinamento.

**Metodologia dos Testes**

Os testes foram realizados somente com uma Rede Neural, haja visto, que a mesma rede já havia sido testada no último laboratório.

O teste realizado fora com a rede tendo 2 Hidden Layers com 32 neurônios cada. Além disso, um learning rate de 0.5 e um batch\_size de 32 imagens. Dessa forma, o treinamento durou cerca de 20 minutos para 50 épocas.



1. **Resultados Alcançados**

Ambos os testes resultaram extremamente parecido com os testes pré-DataAugmentation, porém foi possível notar outras consequências dessa técnica.

Fora visto uma “certeza” muito maior nos testes, demonstrando que a Rede tem mais certeza das informações que estão sendo retornadas.

Além disso, foram testadas ambas a redes com imagens sendo desenhadas manualmente, utilizando a função de Data Augmentation foi possível perceber uma grande melhora para reconhecer dígitos customizados.

Ícone

Descrição gerada automaticamente com confiança baixaLogotipo

Descrição gerada automaticamente

Dessa forma, é possível notar que a rede antiga havia dificuldade em reconhecer o 5, porém utilizando a técnica de Data Augmentation é possível notar uma melhora significativa na certeza e no output da rede.

1. **Conclusão**

É possível concluir que, mesmo que não tenha acontecido uma melhora nos números e dados do treino, foi possível notar uma grande melhora no funcionando da Rede para com dados externos ao DataSet original. É necessário ressaltar que não fora utilizado um banco de dados de validação, somente de treino e teste, talvez se fossem utilizados dados de validação, fosse possível notar uma melhora no treinamento.

Como visto e provado por 50 épocas de treinamento, sem melhora considerável, retornando uma precisão de 93.7%, a Rede se comportou de maneira igual com o DataSet, porém exibindo características melhores para informações externas ao DataSet, também foi possível verificar uma melhora nos valores de certeza das informações atestadas pela Rede.