



MACHINE LEARNING- ATLÂNTICO AVANTI

CLASSIFICAÇÃO DE PLANTAS



EQUIPE 2

- Alisson Chaves
- Ermeson Alves
- Izaquiel da Silva
- Ludmila Shiratori
- Mariana Morales





SUMÁRIO

- Introdução
- Metodologia
- Resultados
- Conclusão
- Referências Bibliográficas





INTRODUÇÃO



O problema em questão trata da classificação de imagens das folhas de diferentes culturas em 38 classes distintas. Essa tarefa é importante porque “as doenças das plantas são a principal ameaça a segurança de alimentos” [1]. O problema representa também um dos maiores desafios nos setores agrícolas.

O conjunto de dados utilizado por nossa equipe é o New Plant Diseases Dataset, uma recriação do **PlantVillage-Dataset** [2]



METODOLOGIA



Para solução desse problema, nos focamos em utilizar o modelo onde foi obtido maior resultado, sendo o **ResNet** que evidencia resultados impressionantes de **99.2%**.

Durante essa análise foi feita uma análise exploratória dos dados o que permitiu a compreensão da quantidade de plantas e variedades de doenças presentes no dataset.



METODOLOGIA



Plant Disease Classification - ResNet- 99.2%

Base de dados:

- New Plant Diseases Dataset



PyTorch

Método: **ResNet**

Análise exploratória de dados utilizado para observar o número de plantas e o número de doenças. Biblioteca PyTorch utilizada. Classificação.



METODOLOGIA

```
'''Configurações e hiperparâmetros'''
random_seed = 2
torch.manual_seed(random_seed)
device = torch.device("cuda:0" if torch.cuda.is_available() else "cpu")
checkpoints_dir = './checkpoints'

batch_size = 32
epochs = 2
criterion = nn.CrossEntropyLoss()

# teste atual: ResNet50
model = models.resnet50(weights=models.ResNet50_Weights.IMAGENET1K_V2)
num_ftrs = model.fc.in_features

# aqui setamos o size para a saída com base no numero de classes
model.fc = nn.Linear(num_ftrs, len(train.classes))
# enviamos o modelo para a GPU, se disponível
model = model.to(device)

optimizer_ft = optim.SGD(model.parameters(), lr=0.001, momentum=0.9)
exp_lr_scheduler = lr_scheduler.StepLR(optimizer_ft, step_size=7, gamma=0.1)
```




ALTERAÇÃO DE PARÂMETROS



optimizer_ft: para uma taxa de aprendizado para 0,0001 por exemplo, e adicionar **weight_decay=1e-4** para evitar que os pesos se tornem muito grandes durante o treinamento

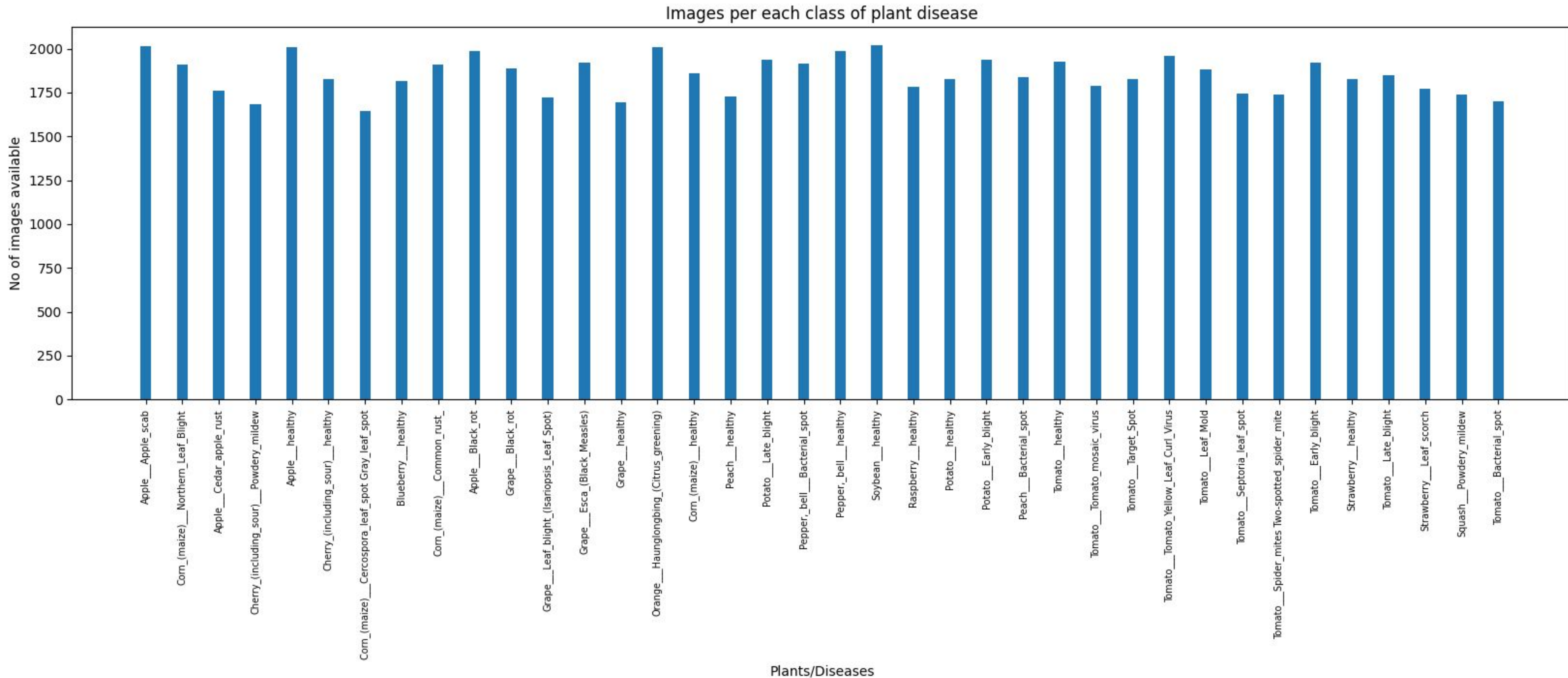
step_size: a gente pode usar a teoria do numero magico de Miller, nesse caso a teoria é $7 +$ ou $- 2$, ou seja uma variação entre 5 ou 9. que é o padrão de aprendizagem.

Segmentação baseada em limiarização:

É uma técnica que busca a separação de objetos ou regiões em uma imagem com base em um valor de limiar (threshold) aplicado a um atributo da imagem, geralmente a intensidade dos pixels.



Número de Imagens por classe de doença/saudável





RESULTADOS



- Uso do **colab** para os resultados preliminares;
- Repositório com o código:
github.com/ermeson-alves/BootcampAvanti-Equipe-2
- Comentários nos códigos das implementações encontradas e adaptação;
- Normalização dos dados;
- Embasamento na **solução com ResNet e 99,2% de acurácia** e no tutorial **TRANSFER LEARNING FOR COMPUTER VISION TUTORIAL**, do **Pytorch**;
- Estratégia de treino com **transfer learning** para os resultados iniciais;
- Função de perda Cross-entropy-Loss, baseada nas aulas;



RESULTADOS

Epoch 0/3 -----

train Loss: 0.5550 Acc: 0.8868

valid Loss: 0.0504 Acc: 0.9862

Epoch 1/3 -----

train Loss: 0.0499 Acc: 0.9865

valid Loss: 0.0246 Acc: 0.9923

Epoch 2/3 -----

train Loss: 0.0250 Acc: 0.9940

valid Loss: 0.0188 Acc: 0.9945

Epoch 3/3 -----

train Loss: 0.0164 Acc: 0.9961

valid Loss: 0.0130 Acc: 0.9963

Training complete in 37m 48s

Best val Acc: 0.996301

predicted: Apple__Apple_scab



predicted: Apple__Apple_scab



predicted: Apple__Apple_scab



predicted: Pepper,_bell__healthy



predicted: Corn_(maize)__Northern_Leaf_Blight



predicted: Potato__Late_blight



predicted: Potato__healthy



predicted: Corn_(maize)__healthy



predicted: Apple__Apple_scab





RESULTADOS

predicted: Grape__Black_rot



predicted: Grape__Black_rot



predicted: Tomato__healthy



predicted: Tomato__Target_Spot



predicted: Corn_(maize)__healthy



predicted: Orange__Haunglongbing_(Citrus_greening)



predicted: Tomato__Early_blight



predicted: Pepper,_bell__healthy



predicted: Corn_(maize)__healthy



predicted: Blueberry__healthy



predicted: Corn_(maize)__Common_rust_



predicted: Potato__Late_blight





CONCLUSÃO



Utilizando uma variante mais simples da ResNet como arquitetura, foi possível obter ganho na perda e na acuracidade de treinamento.

O tempo de treinamento foi de 37m 48s e a melhor acuracidade do modelo foi de 0.996301.



REFERÊNCIAS BIBLIOGRÁFICAS



- [1] MUKTI, Ishrat Zahan; BISWAS, Dipayan. Transfer learning based plant diseases detection using ResNet50. In: 2019 4th International conference on electrical information and communication technology (EICT). IEEE, 2019. p. 1-6.
- [2] "plantvillage_deeplearning_paper_dataset", GitHub repository, 2016, [online] Available: https://github.com/salathegroup/plantvillage_deeplearning_paper_dataset.
- [3] "deep learning book", deeplearningbook, 2022, [online], <https://www.deeplearningbook.com.br/>