Laboratório CNN

Giovanni Rosa (GRR20148144)

Departamento de Informática, Universidade Federal do Paraná (UFPR)

1.Introdução

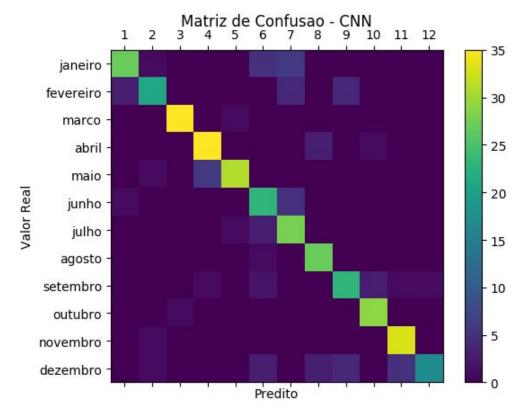
Atividade da disciplina de Aprendizado de Máquina sobre CNN, segundo semestre de 2019.

2. Atividade

Utilizou-se o TensorFlow como backend da biblioteca Keras.

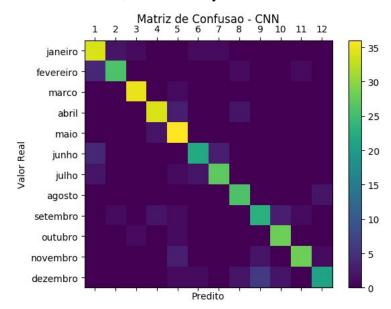
2.1 Avaliando os Parâmetros das Camadas de Convolução

Executar o código com a configuração inicial fornecida gerou os seguintes resultados: Test loss: 50,40%, Test accuracy: 82,04%



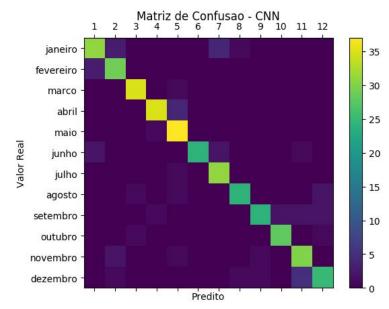
Após calcular a média da altura e largura das imagens e redimensioná-las com esses valores, chegou-se aos seguintes resultados:

Altura média: 60, Largura média: 178 Test loss: 56.92%, Test accuracy: 84.79%



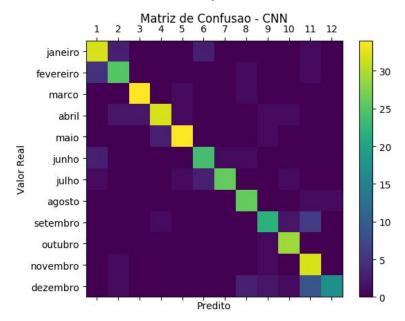
Aumentando o kernel para 5x5:

Test loss: 50.77%, Test accuracy: 88.03%



Utilizando o AveragePooling2D:

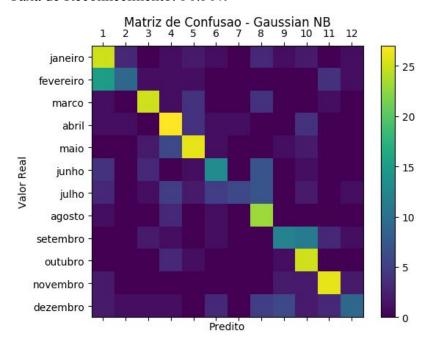
Test loss: 71.55%, Test accuracy: 83.04%



Pudemos notar uma piora dos resultados. Então, a melhor opção é com o MaxPooling2D.

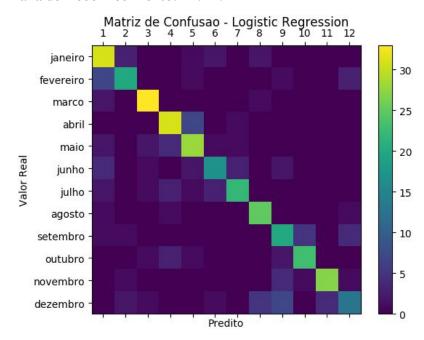
2.2 Características Extraídas via Transfer Learning Usando a Rede Inception

Executando o código com as configurações inicias, ou seja, com GaussianNB: Taxa de Reconhecimento: 56.36%



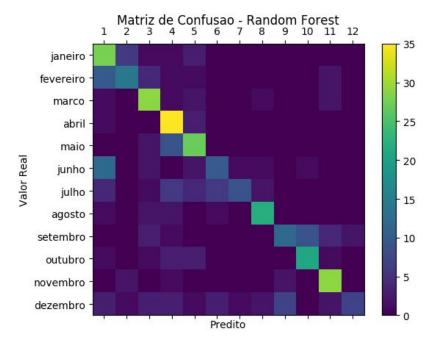
Utilizando o algoritmo Logistic Regression:

Taxa de Reconhecimento: 72.32%



Utilizando o método Random Forest:

Taxa de Reconhecimento: 60.6%

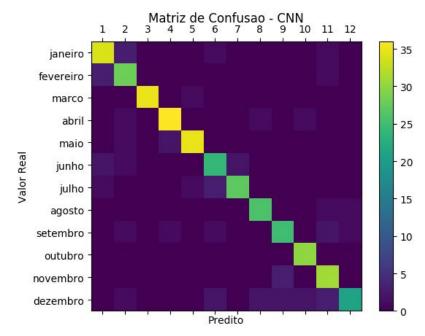


Podemos notar que o Logistic Regression apresentou o melhor resultado. Porém, nenhuma chegou próximo da melhor versão do CNN.

2.3 Data Augmentation

Aplicou-se uma técnica de data augmentation para gerar 3 novas imagens a partir de uma, variando o brilho da mesma. Chegou-se aos seguintes resultados com um tamanho de base de treinamento aumentado para 6312:

Test loss: 61.42%, Test accuracy: 87.78%



Não se pôde notar uma melhora significativa nos resultados, porém o tempo aumentou consideravelmente (de ~10min para ~3h30min). Talvez aplicando uma técnica diferente pudesse ter gerado um melhor desempenho.

3. Conclusão

A técnica de classificação CNN é poderosa e altamente configurável, porém requer um alto poder de processamento. A extração de características por meio do transfer learning não foi bem sucedida. A técnica de data augmentation pode ajudar em casos de uma base de treinamento pequena, contudo o tempo de processamento aumenta bastante. A junção dessas diversas técnicas permite resultados com menos erros. A disponibilidade de tempo e experiência do aplicador também influenciam no momento de fazer escolhas adequadas para montar um bom classificador.

4. Bibliografia

https://www.pyimagesearch.com/2018/12/31/keras-conv2d-and-convolutional-layers/https://keras.io/layers/pooling/

https://machinelearningmastery.com/how-to-configure-image-data-augmentation-when-training-deep-learning-neural-networks/