Phase 1: Preprocessing

The preprocesisng done for this challenge was minimal, given that the Xception was the architecture that showed better performance on the validation set, the respect preprocessing function was used (tensorflow.keras.applications.xception.preprocess_input) just to ensure that the images follow that the distribution that the model was used to take as input.

Phase 2: Training and Hyperparameter tunning

This was phase were I was stuck most of my time during this project. For staters and inseatd of considering just one architecture and further optimize it, I considered several ones (modified VGG-19, modified DenseNet, modified Xception and one designed from the scratch). Promptly I realized that the Xception was the one that achieved the best results on the validation set and begin to do other experiments in order to further improve the models' accuracy and training stability.

I was able to achieve better results thanks to the introduction a attention mecanism that was based on the work of Woo *et al*. - CBAM: Convolutional Block Attention Module, and the tunning of other hyperparameters like the learning rate (0.0006), batch size (32), number of pre-trained initial layers to be extracted from the Xception module (4) and number of final layers (5).

Phase 3: Evaluation

The trained model was able to achieve the following metrics on the test set:

Accuracy: 0.8625

Precision: 0.86

Recall: 0.7414

area under the ROC curve: 0.9105

Despite the good results achieved by the model, it displayed a unstable behaviour with regarding to the validation set. This is a improvement point for the upcoming Kaggle challenges