\*\*\*We performed the main evaluations of our networks on the Bipbip Mais dataset\*\*\*

To create our neural network, at first we started with a "custom" implementation.

For the first part, the encoder that performs the downsampling, we took inspiration from VGG, and we stacked several convolutional, activation pooling layers.

Then after the "bottleneck", we have built the decoder part that performs the upsampling, by using convolutional and activation layers with a decreasing number of filters.

To train the network we performed early stopping techniques on the validation accuracy.

We also used data augmentation; we got the best result by using the larger image size that ram allowed, so that the final upsampling of the mask using nearest neighbours had the least possible impact on the result. After a few trainings we discovered that our custom model reaches 0.30 IoU in test on Bipbip Mais, and we decided to try transfer learning (see next notebook).

Then we have moved to the Transfer Learning technique for what concerning the encoder part.

We have tried two different pre-trained models:

* Vgg16
* Xception

We have freezed the weights of the models and we have added the decoder part.

With the Xception model the IoU was better than the previous one and in test it reaches 0.75.

Then we have tried to ensemble the two models by averaging the prediction of each pixel, but the results were a little bit worse.

After that we have tried to implement a U-Net "by hand", by adding the skip connection between the layers but the results were not better than before.

Finally we have found an implementation of a predefined network similar to U-Net, called Linknet, on github (<https://github.com/qubvel/segmentation_models>), and we used it for transfer learning. We retrained all the weights and we got a final test IoU of 0.82 on Bipbip Mais. To get the results for all the other teams and crops, we just retrained the model for those datasets without particular tuning.