The whole point was to prune the convolution layer based on the most recent activation pooling average throughout the full validation dataset. Therefore, the conv 3 layer needs to be pruned. According to the instructions, we must save the model whenever the accuracy decreases by at least 2%, 4%, or 10%. Model X=2.h5, Model X=4.h5, and Model X=10.h5 are the names of the stored models, which represent drops of 2%, 4%, and 10%, respectively.

To design a GoodNet, we needed to combine two models together which were the BadNet and the repaired model. The program is available in the Homework_2.ipynb file.

Pruned Accuracy Drop	**2 %**	**4 %**	**10 %**
Clean Classification Accuracy	95.744	92.12782	84.333
Attack Success Rate	100.0	99.984	77.209

After training the network with the backdoor, pruning defense is used to eliminate unimportant connections and strengthen the network's resistance to the backdoor. This can be accomplished by ranking the connections in the network according to their importance and removing the least important connections. When the network's performance on the test set is on par with its performance on the clean dataset,

You may gradually prune the network and strengthen it against the backdoor by repeating this process. Afterward, you may use the pruned network as your backdoor detector as it won't be as easily tricked by the covert trigger.

We can observe that the clean classification accuracy drops with increase pruned accuracy drop, but the attack success rate is always higher. Thus, though not completely full proof, this method does make a very resilient model.

Github Repo: https://github.com/sayali1312/ML_for_cybersec_Lab02