

Projet 3:

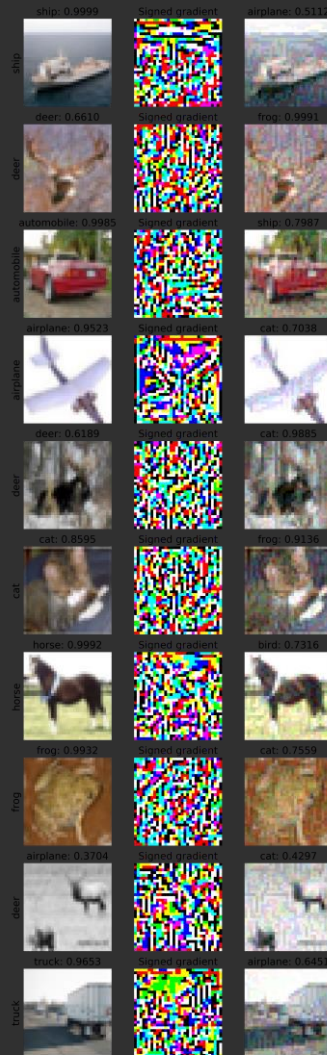
Attaque de réseaux de neurones

Joachim Dublineau, Elie Kadoche, Thomas Petiteau

Sommaire

- ❑ **Attaque FGSM** - Implémentation et performances
- ❑ **Attaque PGD** - Implémentation et performances
- ❑ **Défense de réseau**

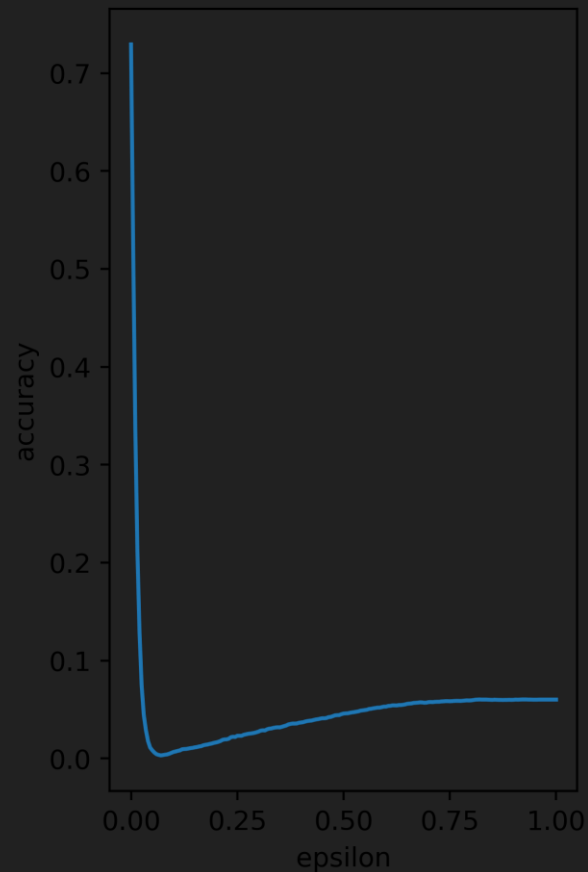
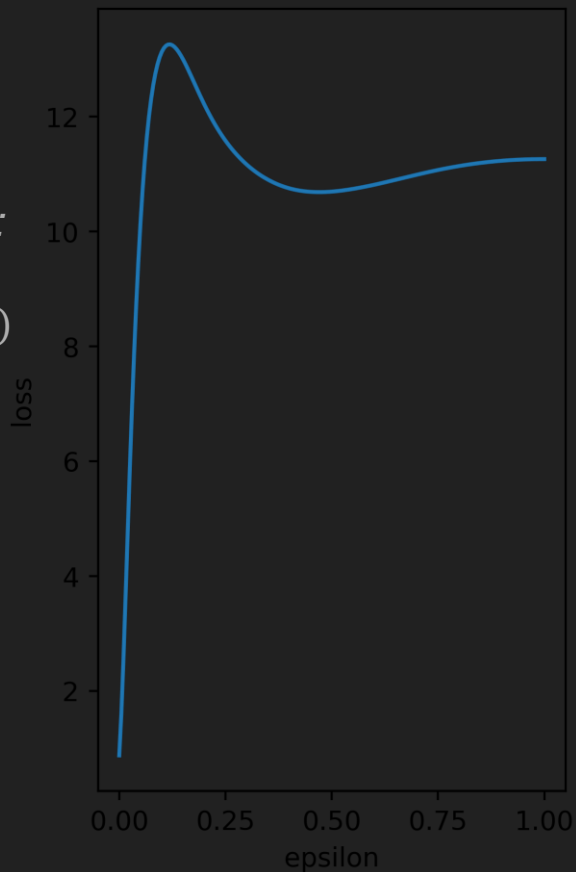
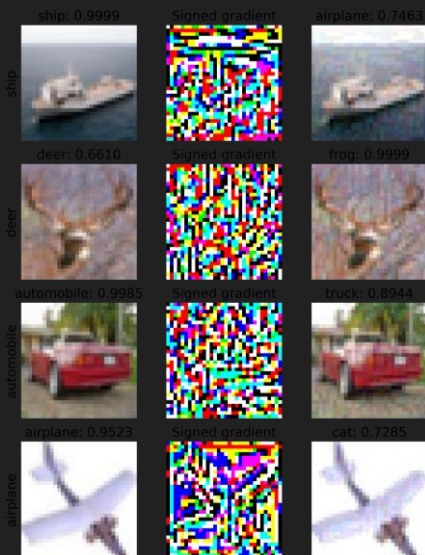
Attaque FGSM



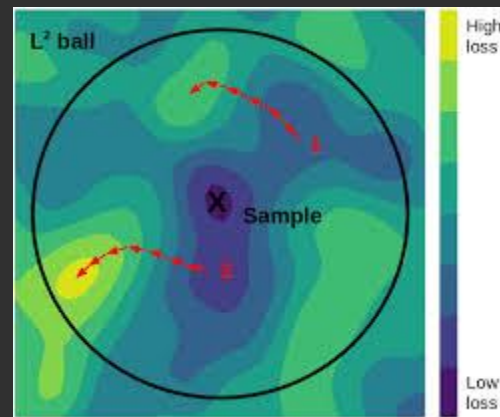
FGSM

Calcul d'une image adversarial:

$$adv_x = x + \epsilon * \text{sign}(\nabla_x L(x, y))$$



Attaque PGD



$$x_{t+1} = \Pi_{\mathcal{B}(0, \epsilon)}(x_t + \eta * \text{sign}(\nabla_x L_\theta(x, y)))$$

Implémentation

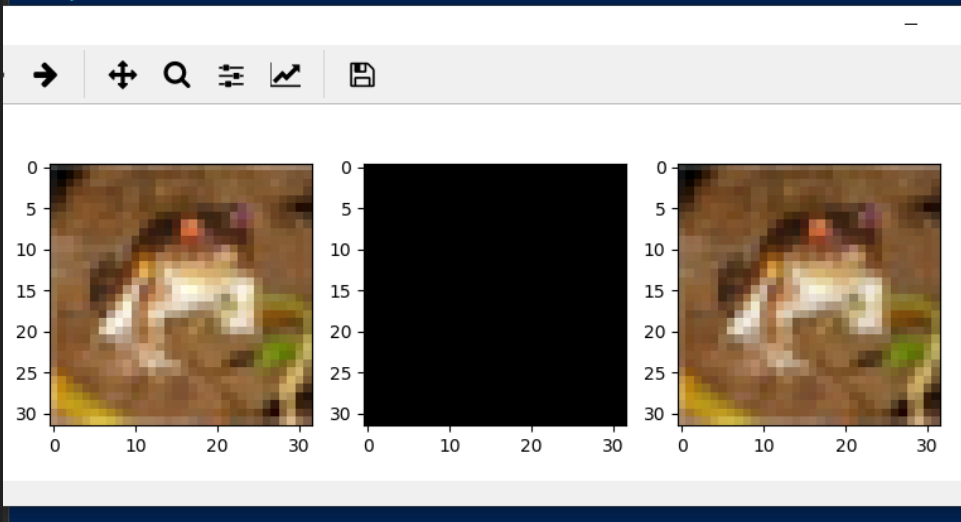
- Calcul du gradient par batch
- Step: $\eta = 0.1$
- Eps = 1
- Condition de convergence: $\|x_{t+1} - x_t\| < \lambda$
 - λ est en général pris à 0.001
- Les attaques sont générées que sur les images bien prédites par le modèle

```
generate_pgd_attacks(model, loss, x, y, eps, batch_size,  
                    step = 0.1, threshold=1e-3, nb_it_max = 20)
```

Performances et résultats

Image:
Norm: 25.16315
Model prediction: 6

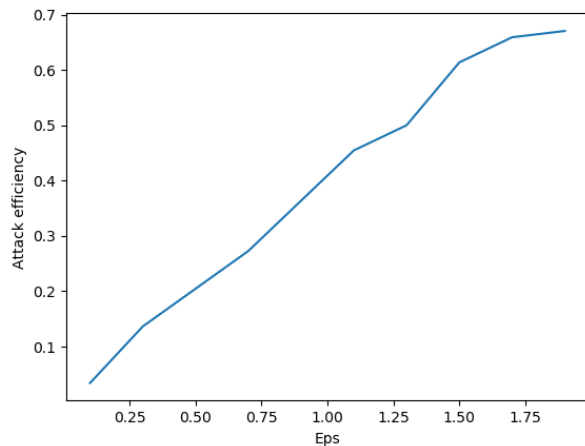
Perturbated image:
Norm: 24.18755
Model prediction: 3



Temps de calcul:

~1sec/image sur i5 2.5 GHz

Efficacité des attaques avec ϵ :



Défense FGSM

Défense FGSM

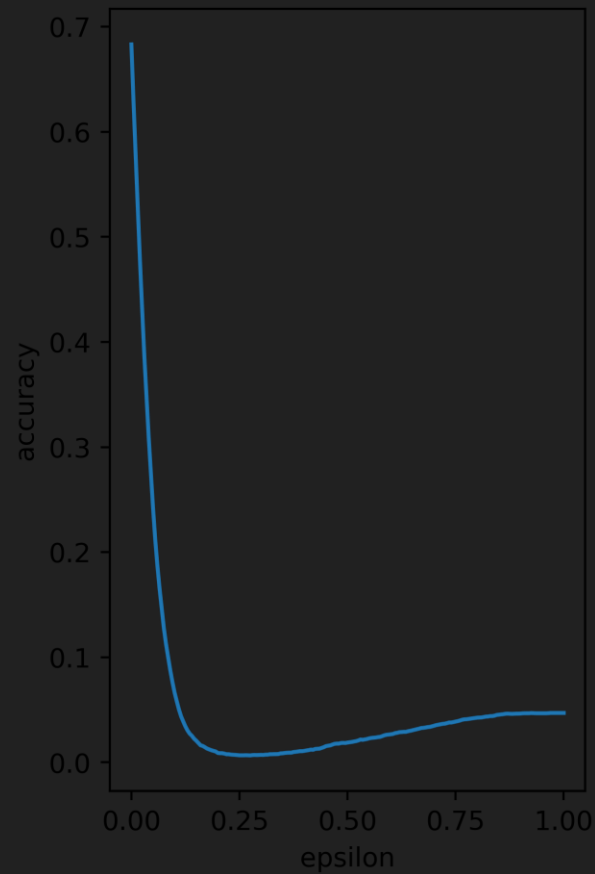
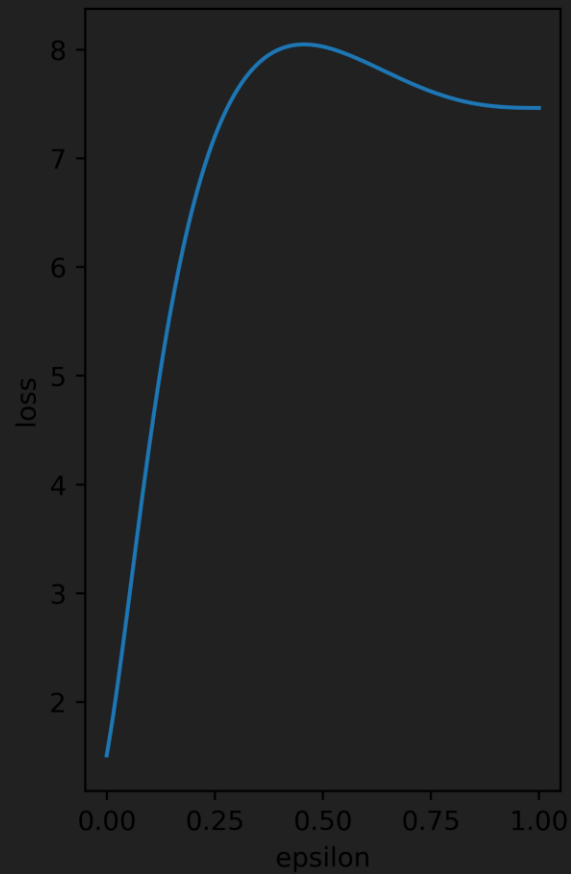
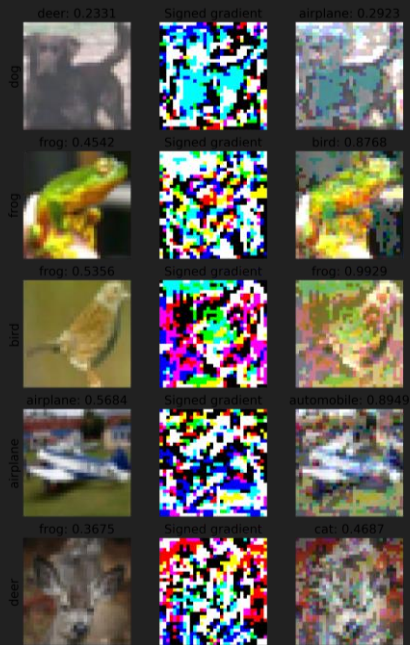
Introduction des images adversariales dans la loss:

$$L(\theta, x, y) = \alpha \times l(\theta, x, y) + (1 - \alpha) \times l(\theta, x + \epsilon * \text{sign}(\nabla_x l(x, y), y))$$

Défense FGSM

$$\alpha = 0.7$$

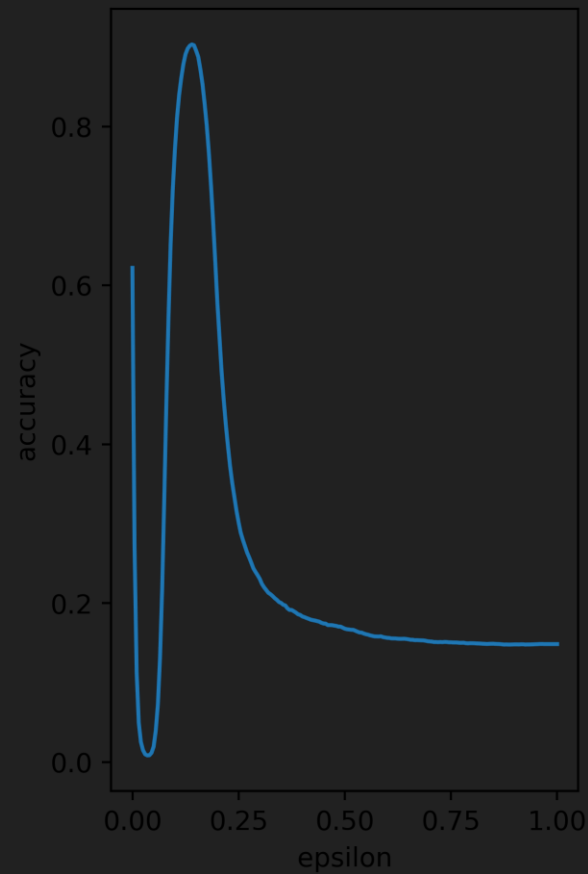
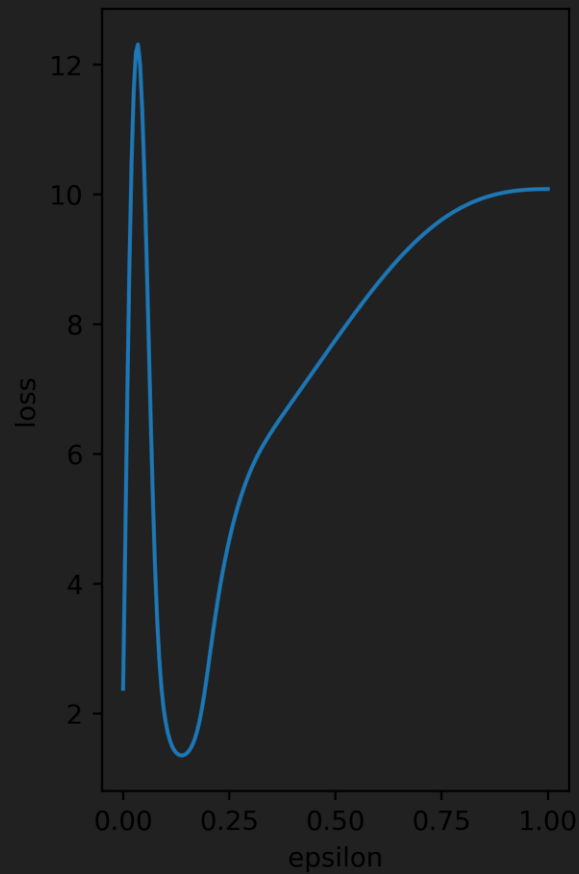
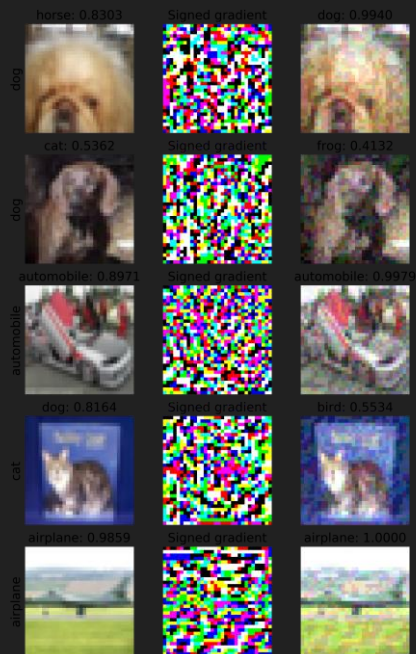
$$\epsilon = 0.025$$



Défense FGSM

$$\alpha = 0.5$$

$$\epsilon = 0.07$$



Défense FGSM

$$\alpha = 0.1$$

$$\epsilon = 0.07$$

