學號：b05901070　系級：電機四　　姓名：蔡昌廷

1. (2%) 試說明 hw6_best.sh 攻擊的方法，包括使用的 proxy model、方法、參數等。此方法和 FGSM 的差異為何？如何影響你的結果？請完整討論。(依內容完整度給分)

在 best 裏頭我先是套用 DenseNet-121 的 model，接著使用 PGD (Projected gradient descent)的方法，此方法會先去計算 loss 對 image 的 gradient 之後，對 gradient 取 sign 函數，找出 gradient 的方向，之後類似 FGSM 的方式將原本的 image 加上 sign(gradient) * alpha，此處的 alpha 算是 learning rate 的效果，為了避免某個 pixel 差距大於原設定的 epsilon，會限制增加的範圍必須介於 –epsilon 至 epsilon 之間，反覆重複這樣的操作(跑多次 iteration)，最後輸出 perturb image。

使用參數：epsilon = 0.1、iteration = 10、alpha = 2/255

FGSM 的方法的目的是要快速的找出 perturb image，根據最初 image gradient 方向直接給定最邊界的新 image，此方法雖然可以迅速找到 perturb image，但其只針對一次 gradient 更新一次 image，無法正確地找到好的 perturb image，因此使用 PGD 多次更新 image 的方法，可以找到在範圍內更佳的 image，使得攻擊成功率可以有效提升，下表為兩者結果。

| | FGSM | PGD |
|---|---|---|
| Success rate | 0.905 | 1.000 |

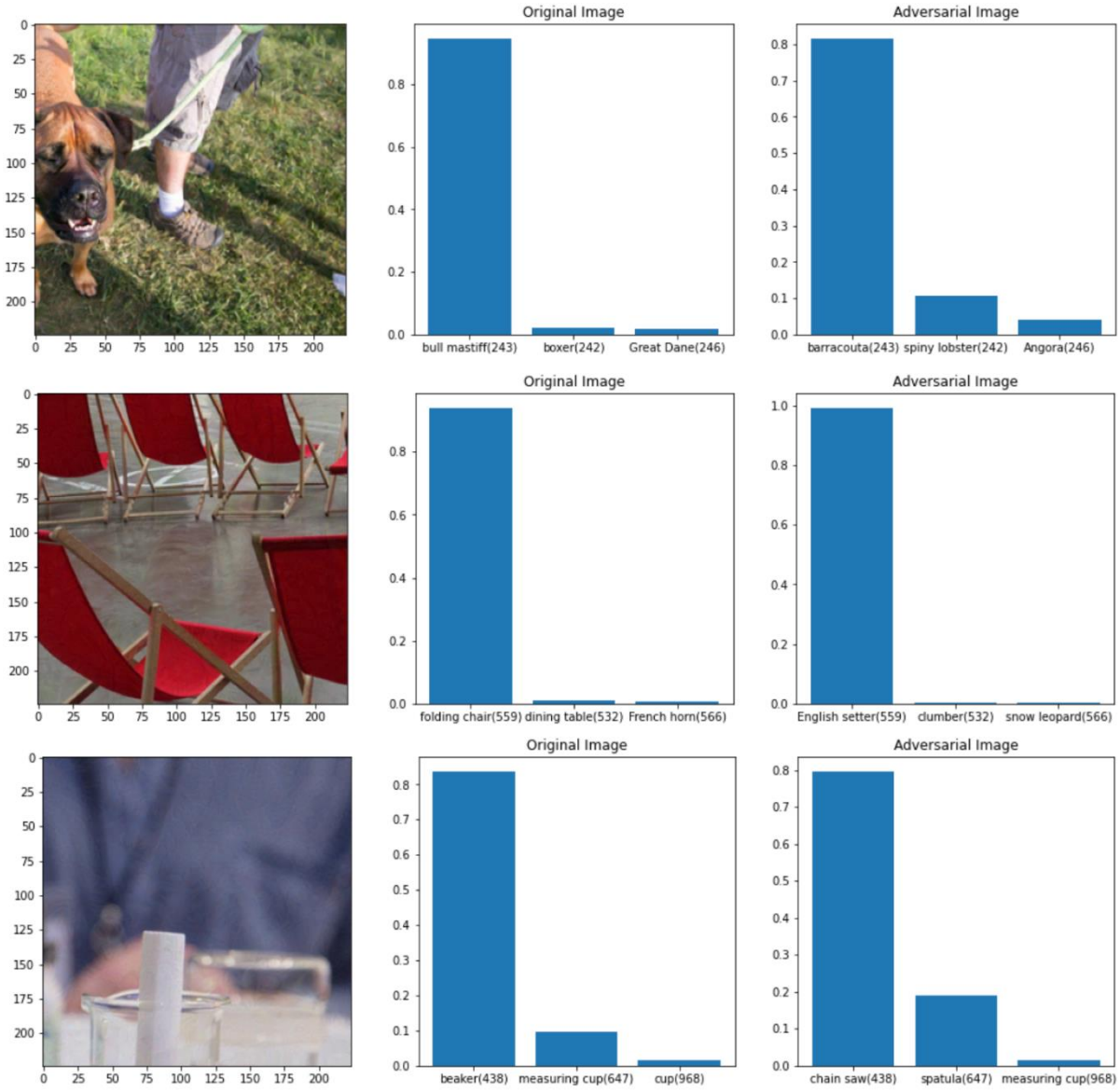
2. (1%) 請嘗試不同的 proxy model，依照你的實作的結果來看，背後的 black box 最有可能為哪一個模型？請說明你的觀察和理由。

以下為用 best attacker 實作不同 model 的結果。

| | Dense169 | Dense121 | Vgg16 | Resnet50 |
|---|---|---|---|---|
| Success rate | 0.610 | 1.000 | 0.315 | 0.48 |
| L-infinity | 5.4900 | 5.5500 | 5.1900 | 6.0000 |

做 black box 攻擊時，如果能正確猜到 black box 之模型，則效果最佳，由上表能明顯看出 Densenet-121 的表現最佳，因此推測背後 black box 的 model 為 Densenet-121，其實在實作的時可以先測試題目不同類型的 net，如 Vgg、Resnet、Densenet，再藉由其表現最佳的再去深入猜測真正的 black box。

3. (1%) 請以 hw6_best.sh 的方法，visualize 任意三張圖片攻擊前後的機率圖 (分別取前三高的機率)。




由上面三張圖可看出，在經過攻擊之後，model 將其辨認出一些奇怪的類別。

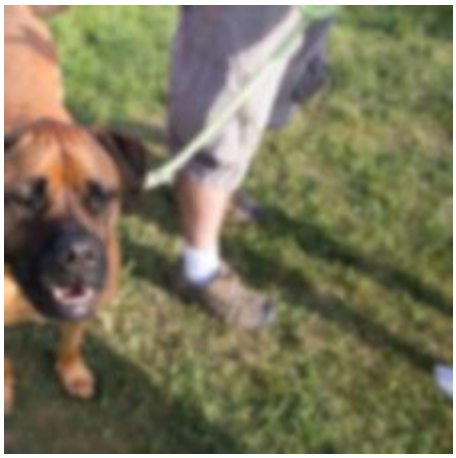
4. (2%) 請將你產生出來的 adversarial img，以任一種 smoothing 的方式實作被動防禦 (passive defense)，觀察是否有效降低模型的誤判的比例。請說明你的方法，附上你防禦前後的 success rate，並簡要說明你的觀察。另外也請討論此防禦對原始圖片會有什麼影響。

此處我選擇使用 Gaussian smooth 來當作 passive defense 的手段，針對每張 adversarial image 用不同 kernel 大小做 Gaussian smooth，再將 image 丟進 Densenet-121 去做 predict，結果如下表

| | 防禦前 | Gaussian(3x3) | Gaussian(5x5) | Gaussian(7x7) |
|---|---|---|---|---|
| Success rate | 1.000 | 0.985 | 0.890 | 0.750 |

經過上表結果發現對 Adversarial image 做 Gaussian smooth 可以些微降低攻擊成功率，我認為最主要的原因是 Gaussian blur 可以降低圖片的雜訊，消除部分帶有攻擊的干擾，而 kernel 越大的 Gaussian blur 自然能消除更多雜訊，缺點是會讓整個圖片更模糊，以下結果為使用 Gaussian blur 實際應用在原始 image 上。

| | 原 image | Gaussian(3x3) | Gaussian(5x5) | Gaussian(3x3) |
|---|---|---|---|---|
| Accuracy | 0.925 | 0.855 | 0.840 | 0.740 |



kernel size = 7

從上面可以發現說如果把 Gaussian blur 的 kernel size 調太高的話，會使得 image 變得太模糊，也有可能影響到原本可以辨識成功的 image，因此在考慮防禦的手段時，也要避免不能讓原本好的 data 因為防禦的關係而出現錯誤。