學號：B07902034　系級：資工二　　姓名：王昱凱

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)

我hw6_best.sh所使用的proxy model為densenet121，而方法為MI-FGSM，也就是momentum iterative gradient sign method，而在epsilon的部分我設定為0.017，iteration為15，執行完的結果l-inf會為1.0，與FGSM的差異在於MI-FGSM會執行比較多個iteration，並且引入momentum進行計算，相較於只進行一次iteration的FGSM，成功率會提升非常多，我最終在JudgeBoi的成功率為0.975，以下為我在網路上所查詢到的MI-FGSM實作方法

Algorithm 1 MI-FGSM

Input: A classifier f with loss function J; a real example $\boldsymbol{x}$ and ground-truth label y;
Input: The size of perturbation ϵ; iterations T and decay factor μ.
Output: An adversarial example $\boldsymbol{x}^*$ with $\|\boldsymbol{x}^* - \boldsymbol{x}\|_\infty \leq \epsilon$.

$\alpha = \epsilon/T$;
$\boldsymbol{g}_0 = 0$; $\boldsymbol{x}_0^* = \boldsymbol{x}$;
for $t = 0$ to $T-1$ **do**
  Input $\boldsymbol{x}_t^*$ to f and obtain the gradient $\nabla_{\boldsymbol{x}} J(\boldsymbol{x}_t^*, y)$;
  Update $\boldsymbol{g}_{t+1}$ by accumulating the velocity vector in the gradient direction as

$$\boldsymbol{g}_{t+1} = \mu \cdot \boldsymbol{g}_t + \frac{\nabla_{\boldsymbol{x}} J(\boldsymbol{x}_t^*, y)}{\|\nabla_{\boldsymbol{x}} J(\boldsymbol{x}_t^*, y)\|_1}; \quad (6)$$

  Update $\boldsymbol{x}_{t+1}^*$ by applying the sign gradient as

$$\boldsymbol{x}_{t+1}^* = \boldsymbol{x}_t^* + \alpha \cdot \mathrm{sign}(\boldsymbol{g}_{t+1}); \quad (7)$$

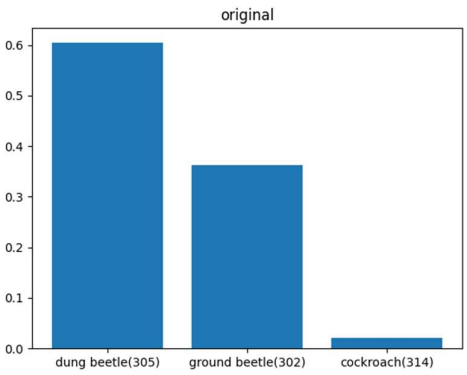
end for
return $\boldsymbol{x}^* = \boldsymbol{x}_T^*$.

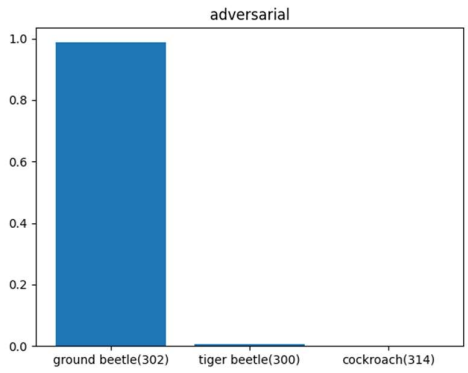
2. (1%) 請嘗試不同的 proxy model，依照你的實作的結果來看，背後的 black box 最有可能為哪一個模型？請說明你的觀察和理由。

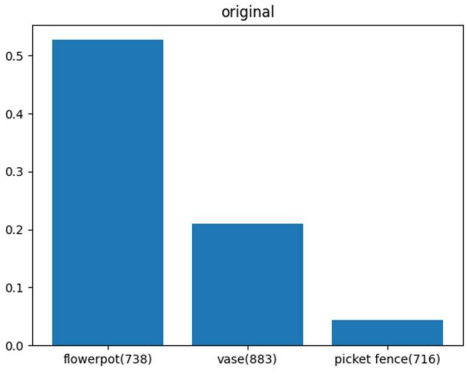
依照我實作的結果來看，背後的black box為densenet121，原因是我在測試fg

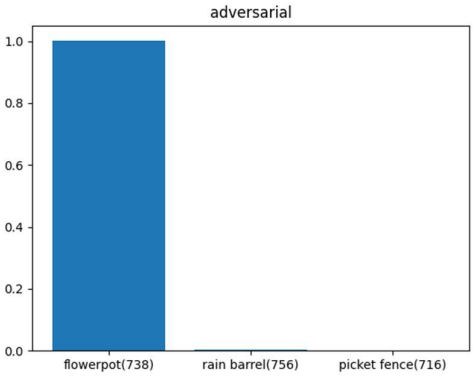
sm的成功率時，我套用每個pretrained model來做測試，並且執行完後上傳到JudgeBoi，只有densenet121有0.8以上的成功率，其餘model都只有0.3, 0.4左右的成功率，因此透過測試可以觀察到block box model就是densenet121

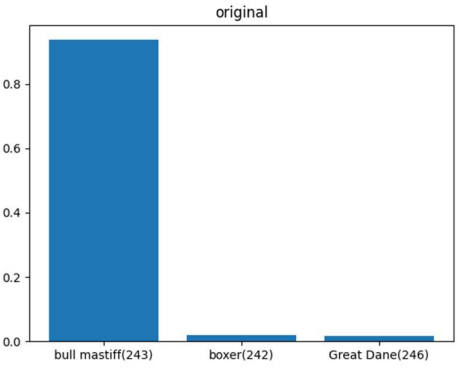
3. (1%) 請以 hw6_best.sh 的方法，visualize 任意三張圖片攻擊前後的機率圖 (分別取前三高的機率)。

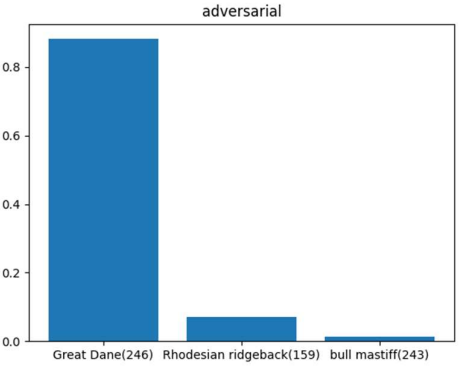












4. (2%) 請將你產生出來的 adversarial img，以任一種 smoothing 的方式實作被動防禦 (passive defense)，觀察是否有效降低模型的誤判的比例。請說明你的方法，附上你防禦前後的 success rate，並簡要說明你的觀察。另外也請討論此防禦對原始圖片會有什麼影響。

   我使用的passive defense方法為gaussian filter，其效果可以模糊並棄除雜訊，而如果是把gaussian filter加在攻擊後的圖片上，攻擊的成功率會從0.975降低至0.7，因此確定此passive defense可以讓某些攻擊失效，而如果是把gaussian filter加在原始圖片上，在model的正確率會從0.925降低至0.73，可知讓圖片變得模糊、平滑，對於model在做分類時也會有一些反效果，從而導致model無法對圖片做出正確的分類