



10-Explainable ML

- 1. 什麼是 Explainable ML ?
- 2. Interpretable vs Powerful
 - 2.1 Interpretable & Explainable
 - 2.2 Decision Tree
- 3. Goal of Explainable ML
- 4. Local Explanation & Global Explanation
 - 4.1 Local Explanation
 - 4.1.1 Which component is critical ?
 - 4.1.2 Saliency Map : 計算梯度
 - 4.1.3 Saliency Map 的局限性 : Gradient Saturation
 - 4.2 How a network processes the input data ?
 - 4.2.1 可視化分析
 - 4.2.2 Probing
 - 4.3 Global Explanation
 - 4.3.1 CNN Filter
 - 4.3.2 Filter visualization
- 5. Outlook

1. 什麼是 Explainable ML ?

要機器給我們它得到答案的**理由**

- 銀行判斷要不要貸款給某一個客戶，但是根據法律規定，銀行作用機器學習模型來做自動的判斷，它必須要給出一個理由
- 機器學習未來也會被用在醫療診斷上，醫療診斷人命關天的事情，也必須給出診斷的理由
- 機器學習模型幫助法官判案，幫助法官自動判案一個犯人能不能夠被假釋
- 自駕車突然急剎時，需要了解它急剎的理由

藉著機器解釋的結果，再去修正模型

2. Interpretable vs Powerful

不使用深度學習的模型，改採用其他比較容易解釋的模型，比如採用 linear model，它的解釋的能力是比較強的。根據一個 linear model 中每一個 feature 的 weight，知道 linear model 在做什麼事

缺點：

雖然比較容易解釋，但 linear model 功能不強大，有很多限制

2.1 Interpretable & Explainable

- **Interpretable**：指模型**不是黑箱**，我們可以容易知道它的內容
- **Explainable**：指模型**是黑箱**，所以必須想辦法賦予它解釋的能力

2.2 Decision Tree

decision tree 相較於 linear 的 model，它是**更強大的模型**，且相較於 deep learning，它**非常地 interpretable**

問題：

光 decision tree 的模型可能不夠強大，一般都是使用數棵 decision tree，也就是 random forest，如此就難以看出其到底如何作出判斷

3. Goal of Explainable ML

好的 explanation 就是人能接受的 explanation，人就是需要一個理由

Make people (your
customers, your boss,
yourself) comfortable.
(my two cents)

4. Local Explanation & Global Explanation



Local Explanation

Why do you think this image is a cat?

Global Explanation

What does a "cat" look like?

(not referred to a specific image)

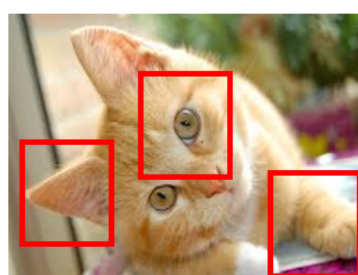
- **Local Explanation**：根據某一個輸入樣本進行回答
給模型一張圖片，模型判斷是一只貓，問模型為什麼覺得這張圖片是一只貓
- **Global Explanation**：根據模型參數本身分析原因

並未給模型任何特定圖片，對具有一堆參數的模型而言，什麼樣的東西叫作一只貓

4.1 Local Explanation

4.1.1 Which component is critical ?

component 可以是圖片的像素、文章的詞匯等等，對每一個 component 做變化、或刪除，如果 network 的輸出有了巨大的變化，就表示該 component 很重要



Which component is critical for making decision?

Object $x \longrightarrow$ Image, text, etc.

Components:

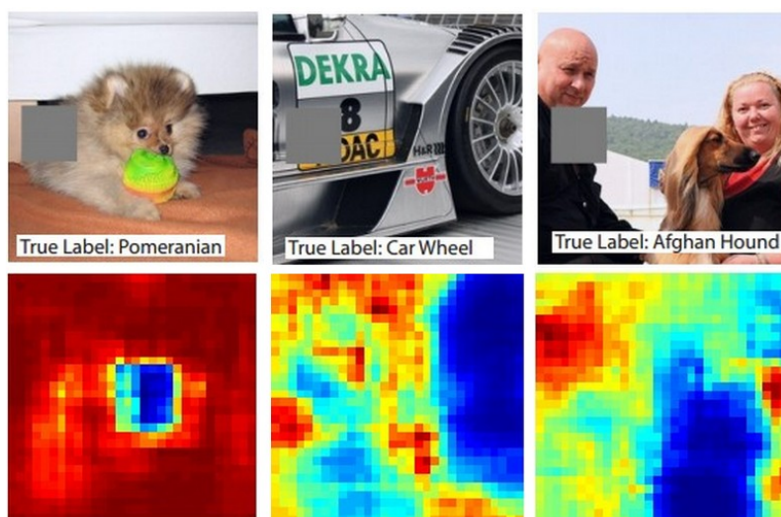
$\{x_1, \dots, x_n, \dots, x_N\}$

Image: pixel, segment, etc.
Text: a word

- Removing or modifying the components
- Large decision change

➡ Important component

在圖片不同的位置放上灰色的方塊，當這個方塊放在不同的地方時，network 會輸出不同的結果



Reference: Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. In *Computer Vision—ECCV 2014* (pp. 818-833)

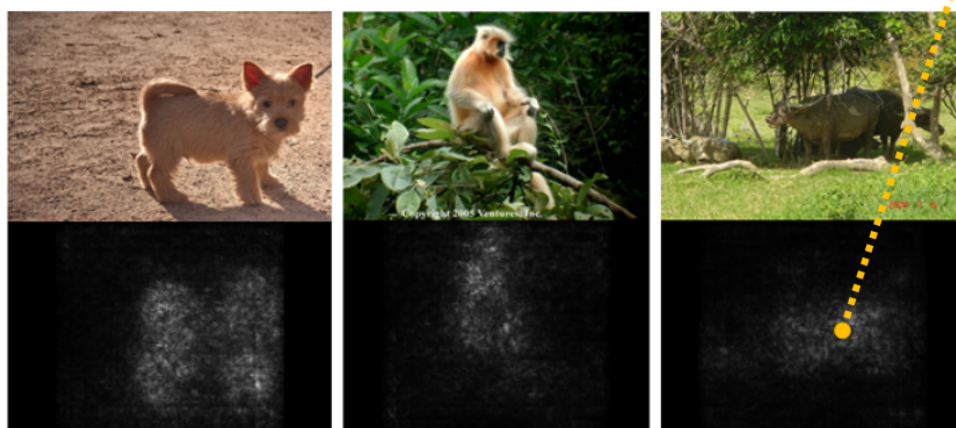
灰色方塊放在紅色區域對輸出結果影響較小，輸出原類別的機率高；但放在藍色區域對輸出結果影響較大，輸出原類別的機率低

4.1.2 Saliency Map：計算梯度

每一個 x 代表一個 pixel， e 是模型輸出的結果跟正確答案的 cross entropy。要考察某一個像素的重要性，可將該像素改變 Δx ，此時 e 也會發生改變 Δe

$$\begin{aligned} \{x_1, \dots, x_n, \dots, x_N\} &\longrightarrow \{x_1, \dots, x_n + \Delta x, \dots, x_N\} \\ \text{pixels} \quad e &\longrightarrow e + \Delta e \\ \text{loss of an example (the difference} & \\ \text{between model output and ground truth)} & \end{aligned}$$

$$\left| \frac{\Delta e}{\Delta x} \right| \longrightarrow \left| \frac{\partial e}{\partial x_n} \right|$$



Saliency Map

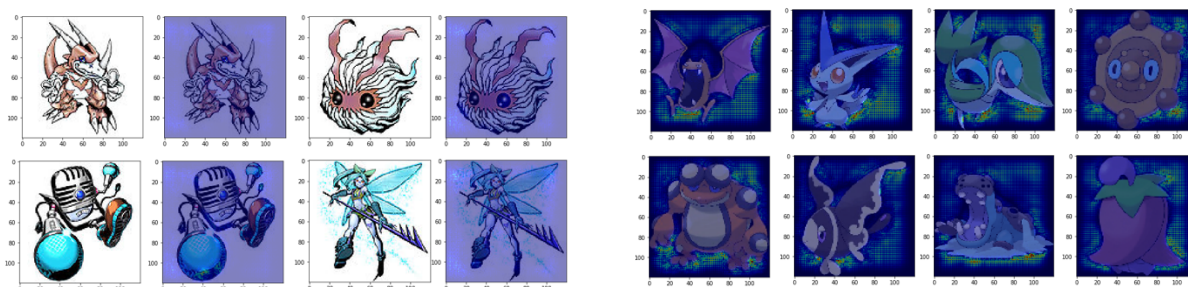
Karen Simonyan, Andrea Vedaldi, Andrew Zisserman, "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps", ICLR, 2014

比值 $\frac{\Delta e}{\Delta x}$ 表示該像素發生改變時，對圖片識別結果的影響，也就是該像素的重要性

對每一像素 x 求偏微分得到的比值都算出來得到 **Saliency Map**

舉例：

對寶可夢和數碼寶貝圖片繪制 Saliency Map，可以看出關注點都是集中在背景，並非物體本身，解釋了 explainable ML 是一個重要技術

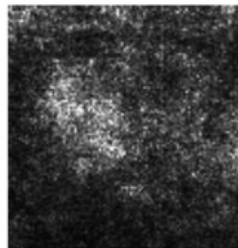


改進 Saliency Map：SmoothGrad

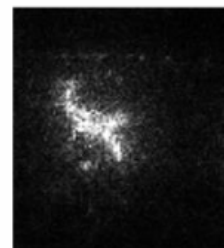
在某一圖片上面加上各種不同的雜訊，得到各種不同的圖片後，對每一張圖片計算 Saliency Map，平均起來得到 SmoothGrad 的結果，如結果往往能夠更加集中在被偵測的物體上



Gazelle
(瞪羚)



Typical



SmoothGrad

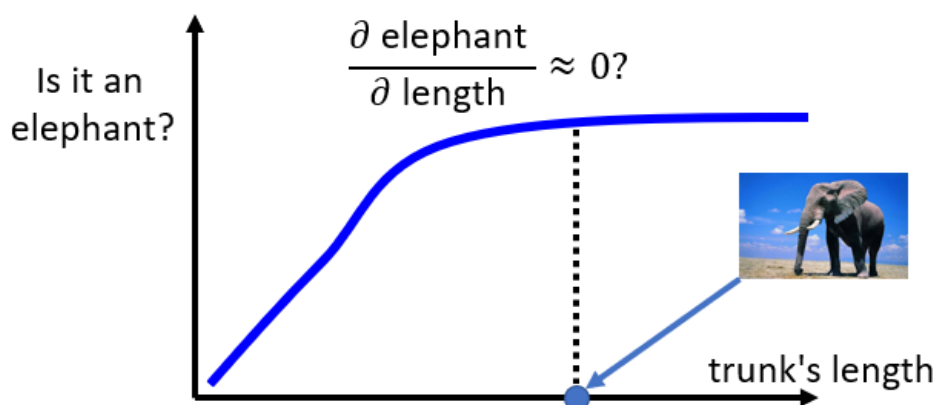
SmoothGrad: Randomly add noises to the input image, get saliency maps of the noisy images, and average them.

4.1.3 Saliency Map 的局限性：Gradient Saturation

當大象鼻子的長度長到一個程度後，就算更長也不會變得更像大象，此時的偏微分為 0

Limitation: Gradient Saturation

Gradient cannot always reflect importance



Alternative: Integrated gradient (IG)

<https://arxiv.org/abs/1611.02639>

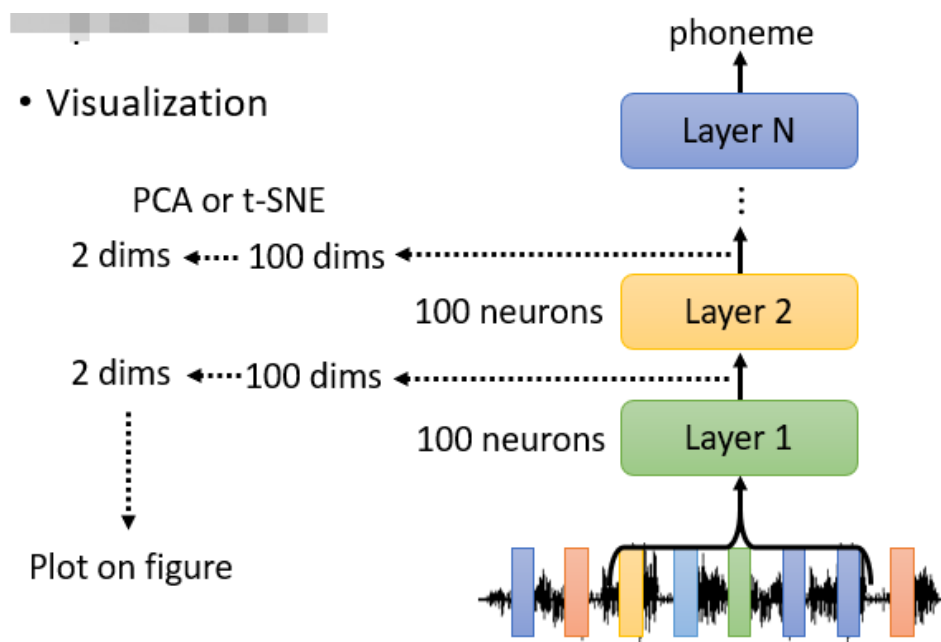
所以光看 gradient 得到的 Saliency Map 就會有錯誤的結論：鼻子的長度，對是不是大象這件事情是不重要的

解決：Integrated Gradient

4.2 How a network processes the input data ?

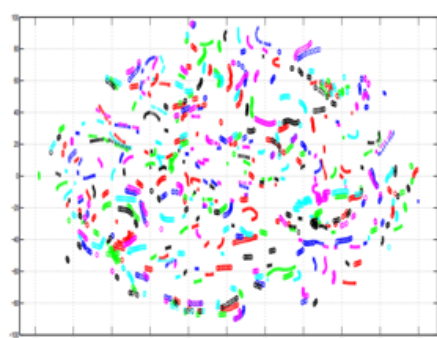
4.2.1 可視化分析

Neural Network :

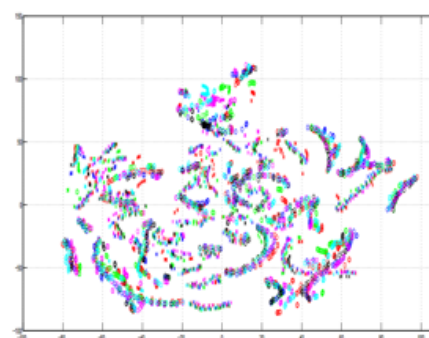


- Visualization
Colors: speakers

A. Mohamed, G. Hinton, and G. Penn,
"Understanding how Deep Belief Networks Perform
Acoustic Modelling," in ICASSP, 2012.



Input Acoustic Feature (MFCC)



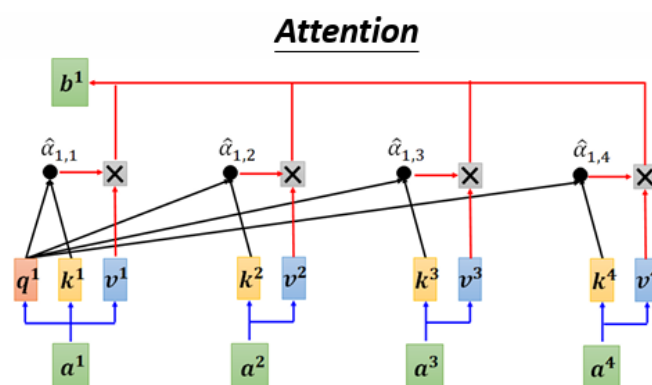
8-th Hidden Layer

對於語音識別問題，取出模型 hidden layer 中的向量作降維可視化。每種顏色代表一個 speaker，可以看到每一個條帶有不同顏色的向量，說明不同 speaker 說類似的話

會被聚攏到空間中的接近位置

Attention :

- Visualization



Attention is not Explanation

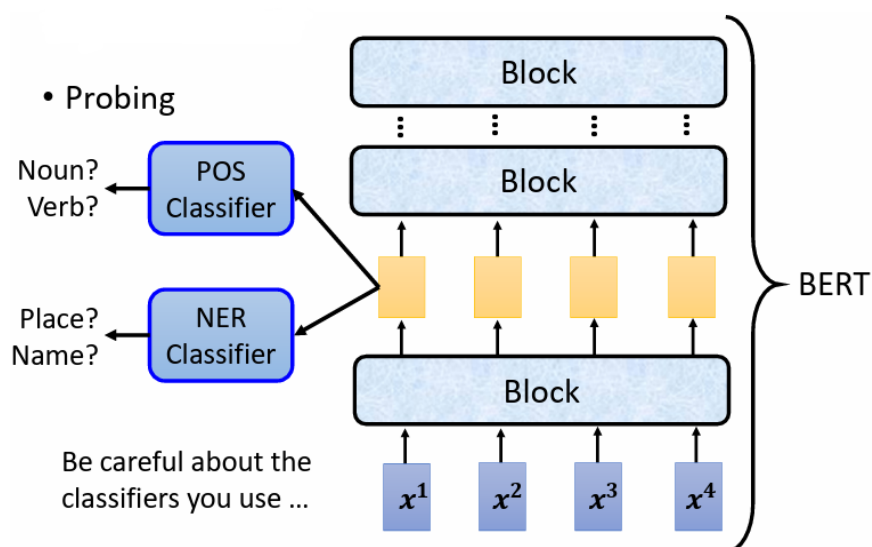
<https://arxiv.org/abs/1902.10186>

Attention is not not Explanation

<https://arxiv.org/abs/1908.04626>

4.2.2 Probing

探測某一層 layer 是否學到了東西



舉例：

1. 對 BERT 進行探測：

訓練 **classifier**，利用模型中的某些層的輸出向量進行一些分類任務，如 POS、NER，正確率高，說明 embedding 有相關信息，反之則無

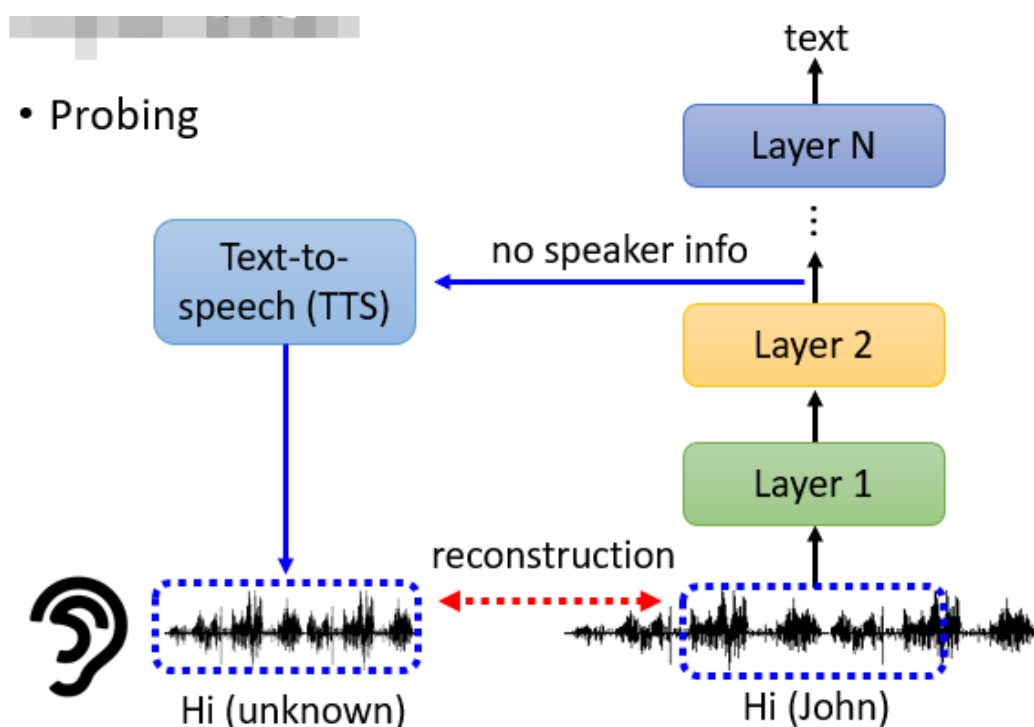
注意：

控制 classifier 的強度。classifier 沒有訓練好，會有可能得出 embedding 沒有相關信息的錯誤結論

2. 對於語音轉文字模型進行探測：

使用 Text-to-Speech 模型，對某一 layer 的向量進行 reconstruction

假設這個 network 做的事是把語者的資訊去掉，layer 2 的輸出沒有任何語者的資訊，這個 TTS 的模型無論怎麼努力，都無法還原語者的特徵，只留下語音的“內容”



4.3 Global Explanation

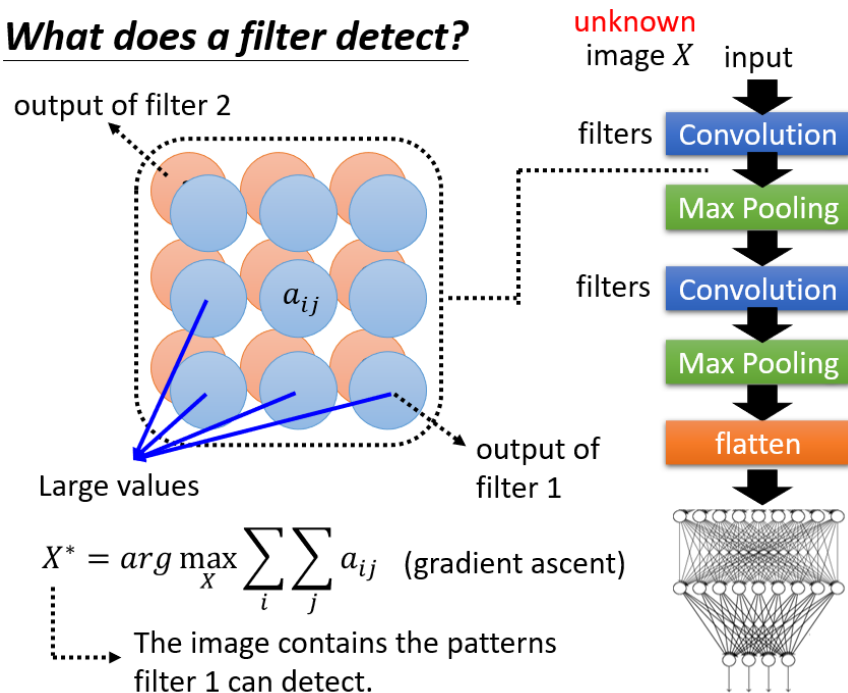
4.3.1 CNN Filter

- **Filter activation**：挑幾張圖片出來，看看圖片中哪些位置會 activate 該 filter
- **Filter visualization**：怎樣的 image 可以最大程度地 activate 該 filter

4.3.2 Filter visualization

以矩陣 X 代表一張圖片，將其當作要學習的參數，解 optimization problem 找出 X^*

What does a filter detect?



方法：

使用 **gradient ascent**

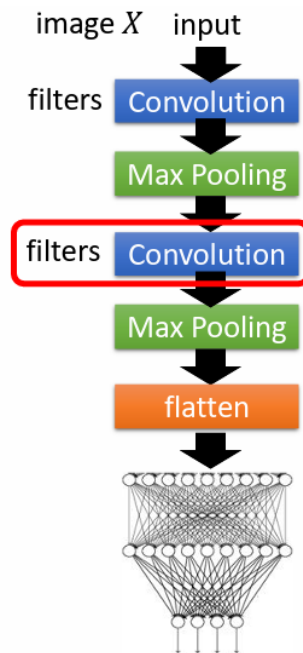
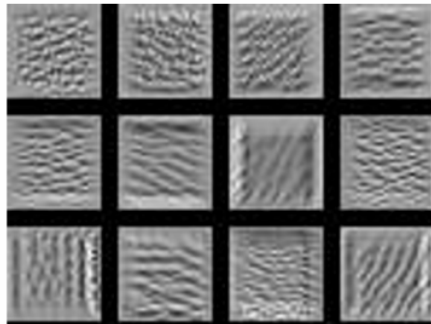
要模型針對每一個 filter 找出一個 X^* ，將它輸入進已經 train 好的 CNN convolutional layer，可以使針對該 filter 輸出的 feature map 中的值越大，說明此 X^* 滿足該 filter 想要偵測的 pattern 的部分越多，由此可以將此 filter 所偵測的 patterns 可視化

舉例：

使用 MNIST 手寫辨識資料集。考慮 filter，找出針對每一個 filter 的 X^* ，並將其可視化

E.g., Digit classifier

X^* for each filter



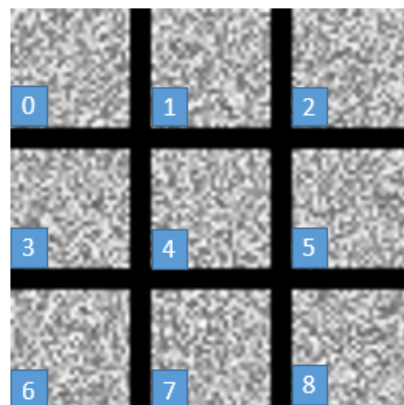
下圖不考慮 filter，而是考慮 image classifier 的最終輸出，期望找出一張圖片 X 輸入進模型後，產生的最終輸出可以讓某一類別的分數越高越好

實驗結果卻是一堆雜訊，沒有辦法看到確切的數字。間接說明了 adversarial attack 中，只需要一些雜訊就能攻擊成功讓模型誤判

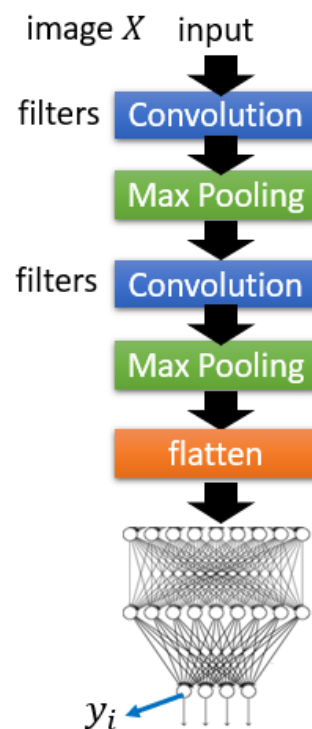
What does a digit look like for CNN?

E.g., Digit classifier

$$X^* = \arg \max_X y_i \quad \text{Can we see digits?}$$



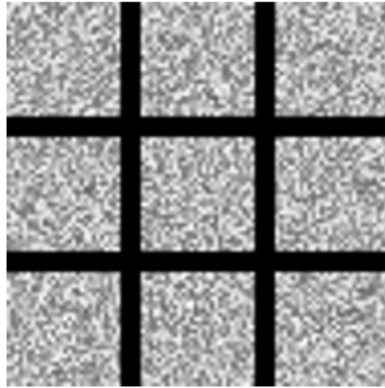
Surprise? Consider adversarial attack!



針對上述方法的 optimization problem 加上一個限制 $R(X)$ ，不只希望輸出的某一類別分數 y_i 要最高，還要考慮 X 多像一個數字，產生的結果可能會好一點

Find the image that maximizes class probability

$$X^* = \arg \max_X y_i$$

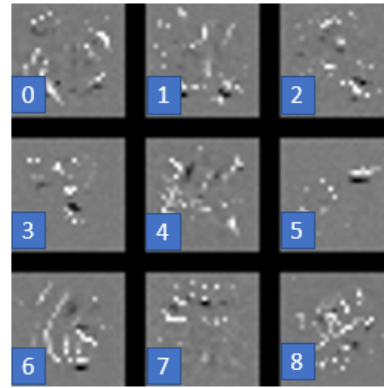


The image should look like a digit.

$$X^* = \arg \max_X y_i + R(X)$$

$$R(X) = - \sum_{i,j} |X_{ij}|$$

How likely
X is a digit



或者利用 **generator** 加上限制，找出 z^* 可以使 y_i 最大化， $X^* = G(z^*)$

• Training a generator

(by GAN, VAE, etc.)



Training Examples

low-dim
vector

z

Image
Generator

G

Image
 X

$X = G(z)$

z

Image
Generator

Image
 X

Image
Classifier

y

$$X^* = \arg \max_X y_i \Rightarrow z^* = \arg \max_z y_i$$

Show image:

$$X^* = G(z^*)$$

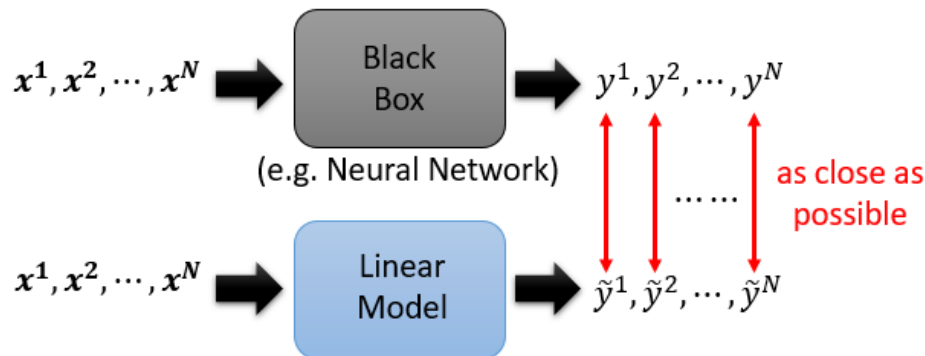
將 image generator 跟 Image classifier 接在一起，image generator 的輸入 z 是從高斯分布採樣得來，輸出是圖片 X ，image classifier 以 X 作為輸入，然後輸出分類的結果 y ，期望 y 對應的某一個類別的分數越大越好

5. Outlook

用一個簡單的可解釋模型模仿一個複雜的不可解釋模型

Outlook

Using an interpretable model to mimic the behavior of an uninterpretable model.



Local Interpretable Model-Agnostic Explanations (LIME)

<https://youtu.be/K1mWgthGS-A>
<https://youtu.be/OjqIVSwly4k>

用線性模型來模仿一個黑盒子模型。由於 linear model 能力有限，不可能模仿整個 neural network 的行為，但可以讓它模仿 neural network 一小個區域的行為，以此解讀那一小個區域裡面發生的事情

方法：

Local Interpretable Model-Agnostic Explanations (LIME)

<https://youtu.be/K1mWgthGS-A>

<https://youtu.be/OjqIVSwly4k>