

Lecture 9: Explainable ML (Machine Learning)

Lectured by HUNG-YI LEE (李宏毅)

Recorded by Yusheng zhao (yszhao0717@gmail.com)

到目前为止，我们训练了很多类型的模型。我们有做到图像分类任务的深度学习模型，输入一张图片，它给我们答案，不单单满足于此，我们还想了解到它得到答案的理由。

Why we need Explainable ML?

- 就算机器总是能得到正确的答案，却不一定表示机器是“智能”的。
 - 神马汉斯的例子
- 现在的人工智能的应用中，可解释性的机器学习模型往往是必须的。
 - 银行用机器模型来判断是否允许放贷，这需要机器给出判断以及给出折服的理由
 - 用机器学习模型来给出医疗诊断
 - 机器学习用在法律上：帮助法官判案、确保机器的公正性，不仅需要答案还需要给出答案的理由
 - 无人驾驶：道路行为例如急刹、加速（决策背后的原因需要合情合理）
- 我们可以基于可解释性的框架来不断改进机器学习模型
 - 无厘头的机器学习：给你的黑箱模型扔一堆数据，经过各种计算，跑出个结果；结果不理想怎么办？爆调参数：lr，改改network架构。——酱紫太不严谨了叭\(\top\)\(\sim\)
 - 但是如果我们知道不理想的模型结果，知道其发生在模型中的具体要素。做到可解释性是非常必要的。

interpretable v.s. powerful

以Linear model为例：interpretable往往要求模型非常简单，这导致相对的not powerful。

以Deep network为例：black boxes...难以解释，但远比Linear model更加powerful

- 不应该因为一个模型difficult to interpretable就扬弃它，而应当再利用powerful的性能的同时去试图追加可解释性。

一位醉汉在路灯下面找钥匙的故事

因此我们的目标就是要找到**强大并且简单到具备可解释性的模型**

- 目标模型1：决策树 (decision tree)
 - decision tree is all you need? :)
 - A tree can still be terrible
 - Kaggle中常用的效果好的技术，不止一棵的decision tree (Random Forest)，实际是若干棵 (e.g. 500棵) 决策树 (森林) 一起起作用，在这种情况下，对于决策作用的解释就比较难说明了。
 - 由此，决策树并不能满足可解释性的机器学习模型这个问题。

Goal of Explainable ML

之前几讲的作业——明确的目标：降低error rate或是提升accuracy

然而，Explainable ML的目标是不明确的。（作业也没有leaderboard）关于Explainable ML的目标以下几点看法：

- Completely know how an ML model works? (整个模型在做什么事，完全了解模型如何做一个决策)
- 我们真的需要Completely know how an ML model works? 事实上，我们也不完全了解人脑是如何运作的，但是我们完全相信人所做出的决策。
- 一个有趣的心理实验

The Copy Machine Study (Ellen Langer, Harvard University)

“Excuse me, I have 5 pages. May I use the Xerox machine?”

60% accept

“Excuse me, I have 5 pages. May I use the Xerox machine,

because I'm in a rush?”

94% accept

“Excuse me, I have 5 pages. May I use the Xerox machine,

because I have to make copies?”

93% accept

- 好的Explanation：人能接受的Explanation

玄学的心理要素：给一个“理由”，让人（用户、reviewers、自身）comfortable，让人高兴。

Explainable ML

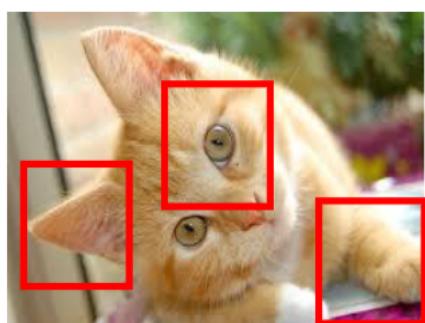
分成两大类：**Local Explanation**、**Global Explanation**

Local Explanation

对特定的某个数据要求机器（模型）做一个针对结果的解释

Why do you think this image is a cat?

Which component is critical?



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

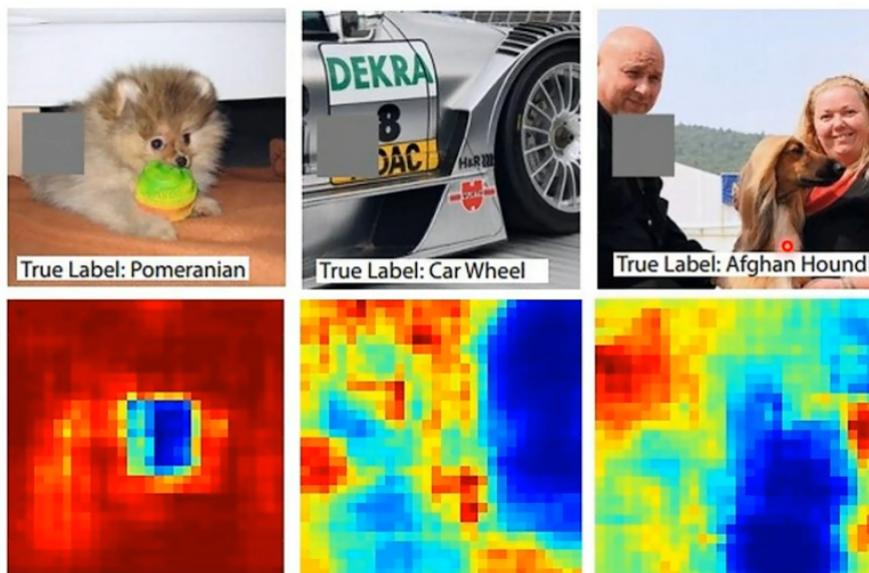
如上图，给机器一张图片的时候，图片里的什么东西（eye? ear?）让机器觉得这是一只猫（做出判断）

对于Object x ，推广来说可以是影像、文本等，这个 x 可以拆成若干个 component

$x = \{x_1, x_2, \dots, x_n, \dots, x_N\}$ ，对应来说这里每一个 component 就是一个 pixel 或者 segment 或者 a word (token) 等——这些 component 中哪些/哪一个对于机器做出判断起到决定性作用？

(最简单的一种方式) 类似于对照实验：把每一个component单独拿出来，做一个改造/删除，之后如果network的输出发生巨大的变化，那么表面这个component必不可少。

E.g. 用mask盖住图像来测试网络的输出 (控制变量了属于是)



这个实验提供了网络中对于目标的位置信息。

量化component重要性：进阶的一种方式（计算梯度）。

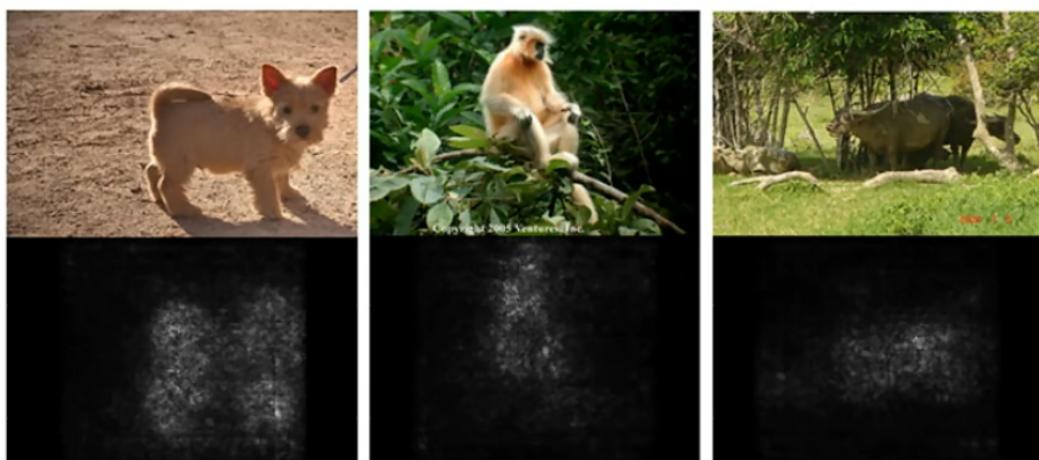
有一张图片将其写做 $\{x_1, \dots, x_n, \dots, x_N\}$ ，每个 x_i 代表一个pixel。

- 然后我们来计算这个例子的loss，记作 e ，这个loss值是ground truth和网络/模型输出的相似性的度量（一般是cross entropy）。 e 越大表示辨识结果越差。
- 对于某一个pixel，将其component加一个小小的增量 $\{x_1, \dots, x_n + \Delta x, \dots, x_N\}$ ，计算所得的 $e' = e + \Delta e$ ，如果这个 Δe 很大，表明这个component对网络输出有着较大的影响。如果 e' 趋近于 e ，那么这个pixel就比较无关紧要。我们通常用

$$|\frac{\Delta e}{\Delta x}| \rightarrow |\frac{\partial e}{\partial x_n}| \quad (1)$$

来表示相对应component的重要性（好像灵敏度分析），以上公式表明该测量其实就是 x_N 对loss做偏微分。

- 把这张图片中每一个pixel的这个重要性都算出来，组成新的图称之为**Saliency Map**

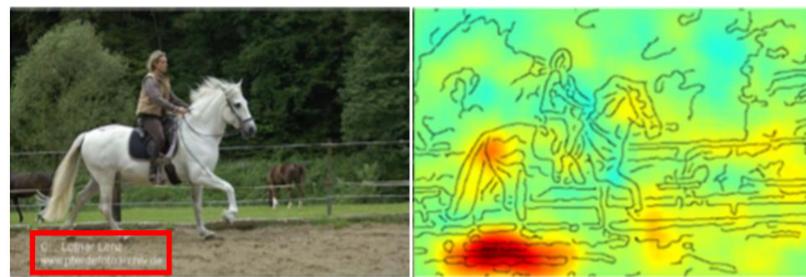


图上越偏白色，表明值越大，这个pixel越重要。

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

- 实践：训练一个分类宝可梦和数码宝贝的分类器

- 资料:
 - Pokémon images: <https://www.Kaggle.com/kvpratama/pokemon-images-dataset/data>
 - Digimon images: <https://github.com/DeathReaper0965/Digimon-Generator-GAN>
- 实验结果，泛化Accuracy十分amazing!
- 画Saliency Map
 - 资料差异。画的map发现机器只关注非本体部分（背景），实际上因为文件格式差异，宝可梦和数码宝贝资料图像背景颜色不同（透明即黑。机器只根据这个来实现二分类。
- 更多的实践:
 - PASCAL VOC 2007 data set



This slide is from: GCPR 2017 Tutorial — W. Samek & K.-R. Müller

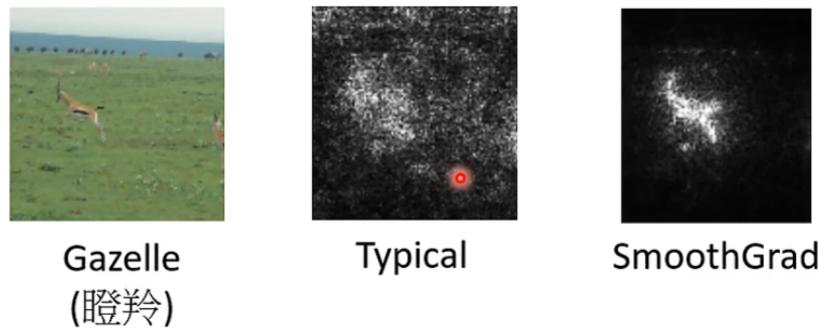
数据集的巨大影响：机器的关注点。判断“马”的训练资料通常有相同的水印，水印部分便成为判断是否马的重要的component。

↑的启示：Explainable Machine Learning是非常重要的。

Limitation: Noisy Gradient

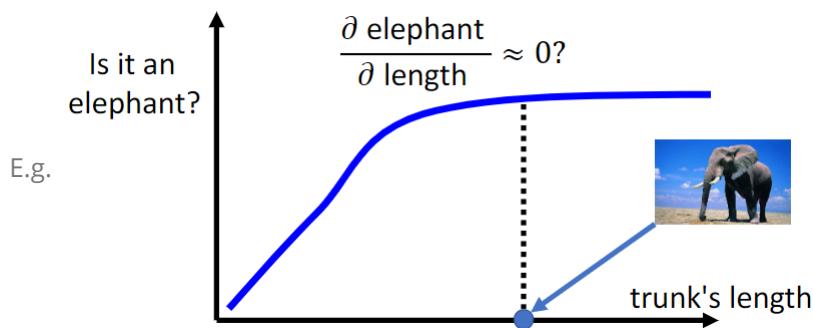
有没有什么方法能把Saliency Map画得更好呢？

- 方法一：SmoothGrad



流程：在图片上面加上不同的噪声，得到若干张不同加上噪声的图片，从而获得数张Saliency Map，平均起来就得到SmoothGrad的结果。

光看梯度（Gradient）并不能反映component的重要性或者说梯度（Gradient）并不能总是反映component的重要性



鼻子越长，大象的可能性趋近不变，这是得到的偏微分趋近于0，难道这就说明了鼻子长度的变化与是否大象可能性的变化无关？显然是错误的。

所以光看偏微分的结果没法完全反映component的重要性

改善的方法：Integrated Gradient (IG) : <https://arxiv.org/abs/1611.02639>

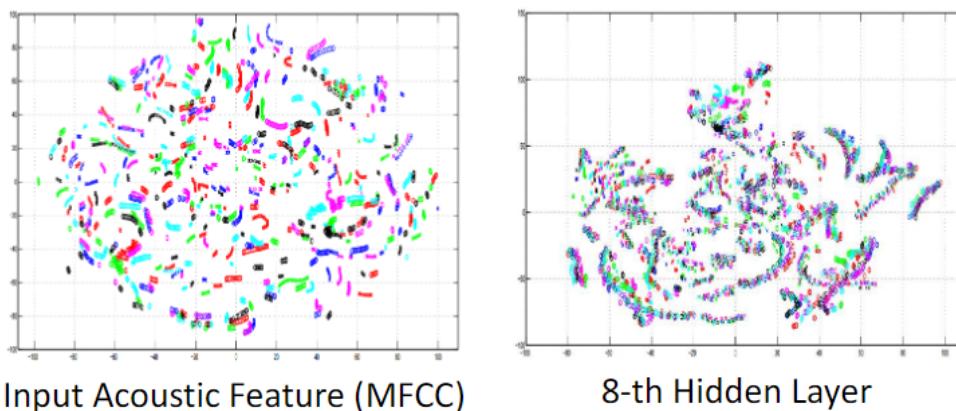
How a network processes the input data? ——network是怎么处理这个输入的呢

- 最直觉的：可视化 (Visualization)

作业BERT以及语音识别，

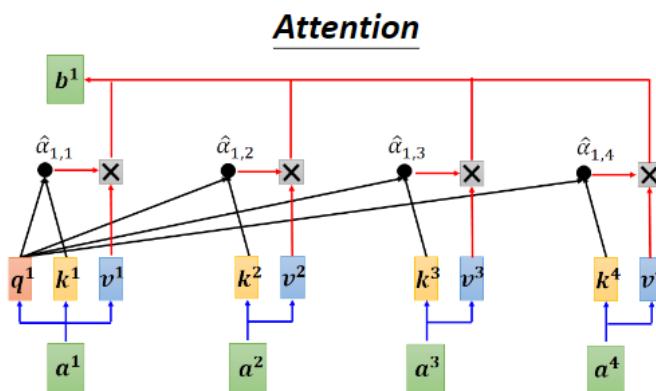
- 以一个layer的100个神经元为例：PCA or t-SNE让100维向量降到2维；plot on figure

Colors: speakers



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

- 分析attention的layer



一些文献：[Attention is not Explanation](#)、[Attention is not not Explanation](#)

"attention能不能被解释"依然是尚待研究的问题 (禁止套娃蛤)

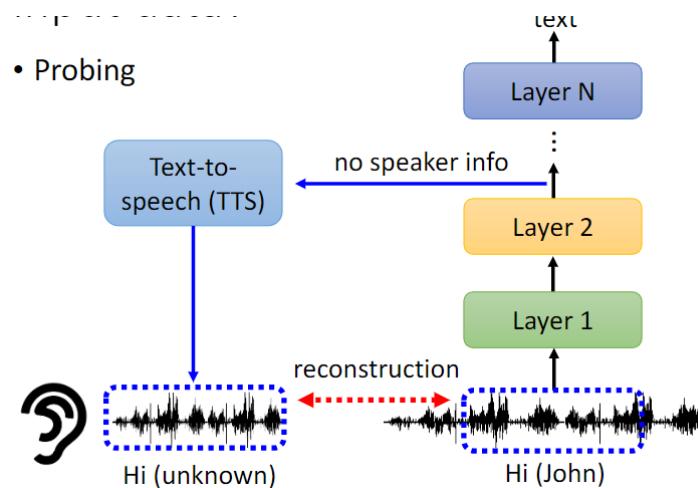
- 另外一种技术：探针 (Probing)

用探针插入network看看发生了什么事，举例来说，想了解BERT的layer里面到底学到了什么东西，实际上可视化技巧就比较局限了。

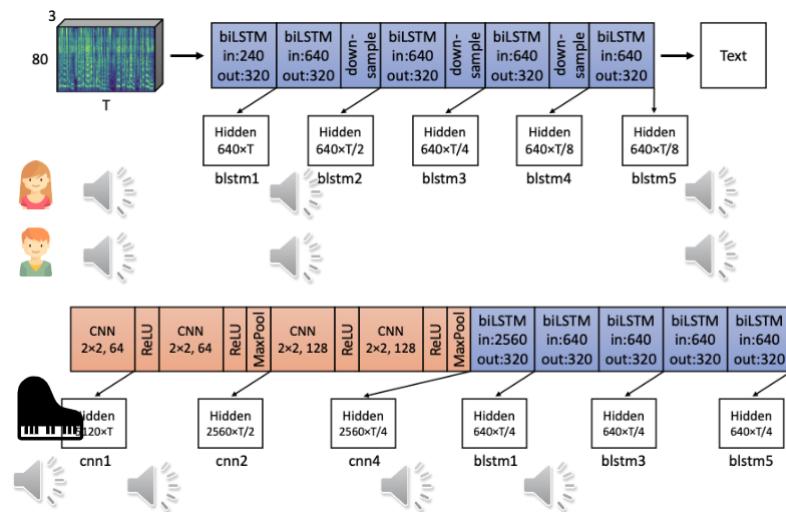
所以我们可以使用探针——实质上是训练好的一个分类器：(E.g.)

- 探针1：POS Classifier——根据一个feature即输入的向量（Embedding）判断其POS tag，也就是来自于哪一个词性的词汇。如果正确率高，则表明其有丰富的词性的资讯；如果正确率低，则表明这些embedding没有词性的资讯。
- 探针2：NER Classifier（命名实体识别Named Entity Recognition，简称NER），他根据这些输入的feature，判断哪些词汇是属于人名还是地名，还是说其他专有名词。
- Tips：小心使用的classifier的强度，如果这个classifier太烂了（没train好），就会影响关于embedding资讯的判断...

关于probing的例子：训练一个语音合成的模型（训练这样一个TTS【Text to Speech】模型，把某一个layer的embedding吃进去，尝试“重建”原始输入），我们可以通过重建后的输出了解到——如果说里面一层layer除去了语者的声音特征，只保留了声音讯号内容所完成了这样一个流程。



另一个例子：5层的biLSTM。声音讯号作为输入，输出一段文字。语音辨识模型。男生和女生的声音资料在通过5层的biLSTM后声色没有什么区别了（人耳无法区分）。

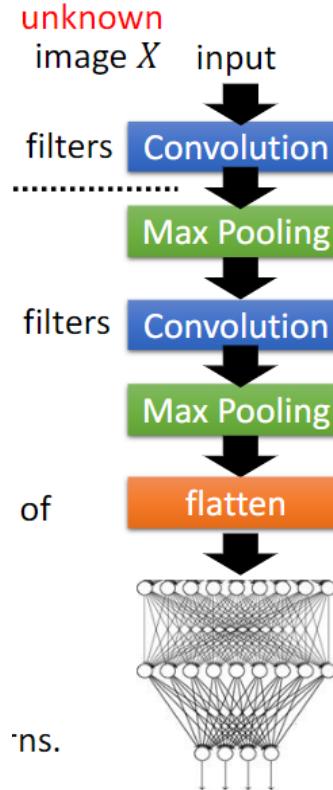


上图另一个例子，3层的CNN和3层的BiLSTM。输入有背景音，在第一层LSTM滤掉了大半背景声。

Global Explanation

以图片分类为例，假定我们还没有开始对数据集进行classify，我们需要对classify整个model的参数特征做出解释（比如说什么样的东西可以是一只猫，如果分类任务中包含猫咪的话），对一个network而言一只猫应该长什么样子。而不是针对指定数据（点/图片）的进行分析或结果。

E.g.假定已有一个train好的CNN，里面有若干层卷积层（Conv Layer），有一堆filter。一张图片作为输入，conv layer输出一个feature map，那每一个filter都会给我们一个metric



假设输入一张图片 X ，通常是一个矩阵。把图片丢进这个CNN里边，我们会发现某一个filter（假设filter1）在它的feature map里面有几个位置有比较大的值（large values）——意味着这个图片里有很多的pattern是由filter1负责侦测的。

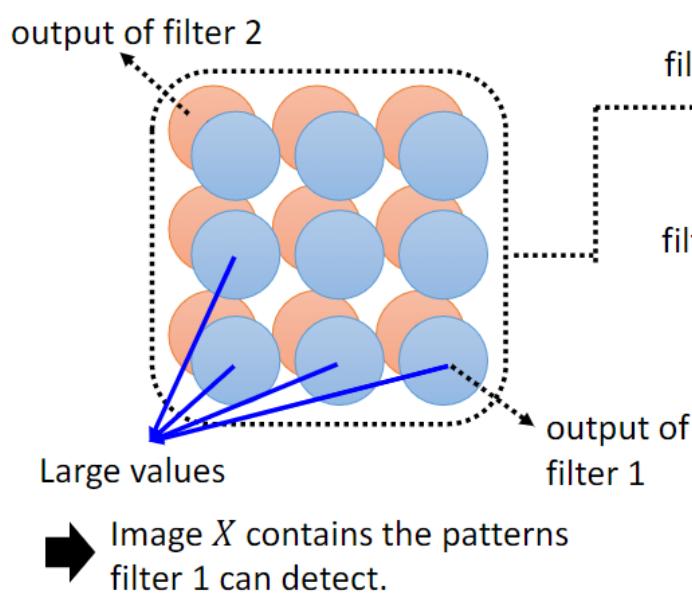
我们想要知道的是对于filter1而言，其理想的pattern的feature map究竟长什么样子——怎么做？答：我们/机器可以创造出一张图片

filter1的feature map是一个矩阵，矩阵里每一个元素记作 a_{ij} ，我们把要找的那张图片 X 当作一个未知变量，当作我们要训练的那个参数，如上图所示，这个未知变量丢进CNN以后，我们考察的filter的位置所输出的feature map理想情况下矩阵的 a_{ij} 总和要越大越好。综上，满足

$$X^* = \arg \max_X \sum_i \sum_j a_{ij} \quad (2)$$

这个 X^* 不是数据集里面一张特定的图片；我们把 X^* 丢进CNN中，看filter1输出的feature map，值越大越好。（原理：gradient ascent）

What does a filter detect?



Let's **create** an image including the patterns.

unknown
image X input

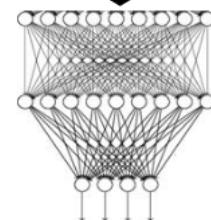
filters Convolution

Max Pooling

filters Convolution

Max Pooling

flatten



以手写数字分类器为例 (digit classifier)，我们按照以上方法想找到网络中间的filter的理想feature map长什么样，也可以看到网络最后的output (令各自分类置信度最高)。

What does a filter detect?

E.g., Digit classifier

X^* for each filter

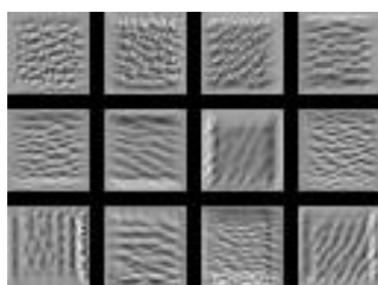


image X input

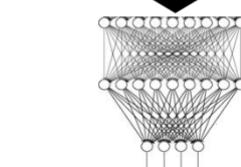
filters Convolution

Max Pooling

filters Convolution

Max Pooling

flatten

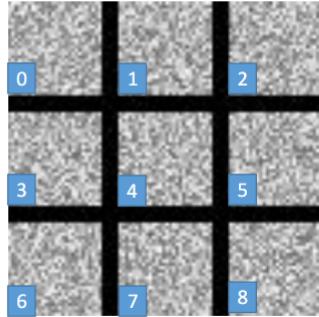


实际上，filter确实表达出其想看到的feature，例如横线、直线、斜直线等

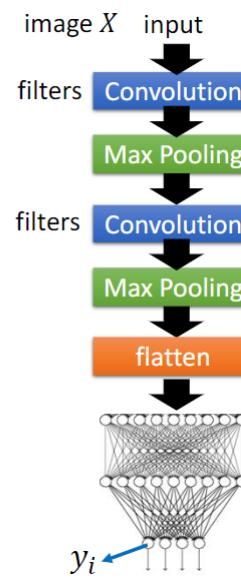
What does a digit look like for CNN?

E.g., Digit classifier

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



Surprise? Consider adversarial attack!



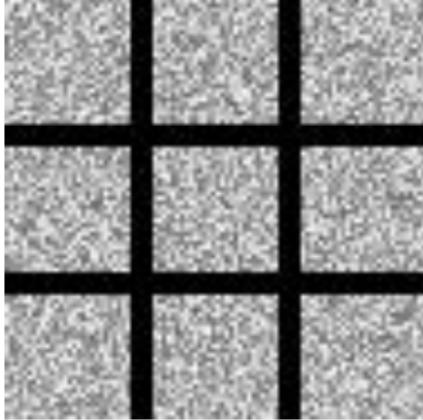
可是，对于分别设置高置信度maximum出来（创造出来的）图片 X^* ，看起来实在没什么区别！

如果我们想要图片 X^* 得到人可以想象（肉眼识别的）数字图像，我们需要加一点限制。举例来说，我们加一个对数字的期望 $R(x)$ ，这个 $R(x)$ 表示how likely X is a digit，这里的

$$R(x) = - \sum_{i,j} |X_{ij}| \quad (3)$$

Find the image that
maximizes class probability

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

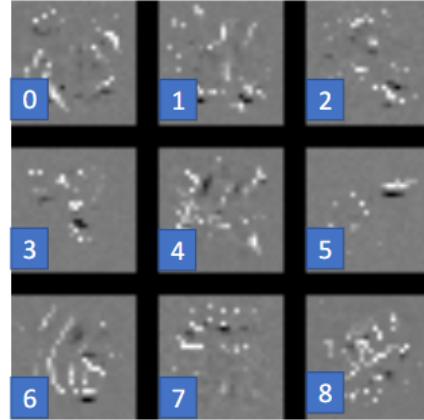


The image should look like a digit.

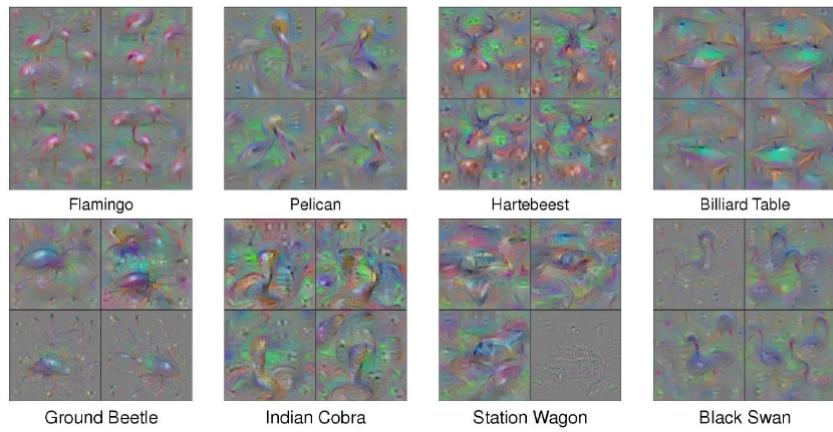
$$X^* = \arg \max_X y_i + R(X)$$

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

How likely
 X is a digit



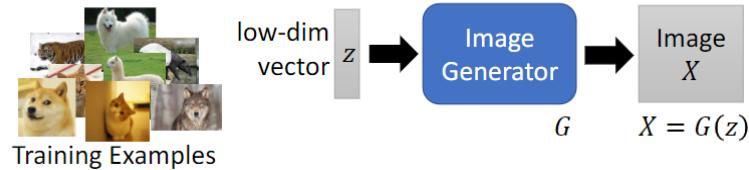
在文献<https://arxiv.org/abs/1506.06579>，爆调超参数，各种正则化.....“反推”得到



有效的一招: Constraint from Generator

- train一个image的generator, 可以用GAN or VAE

- Training a generator (by GAN, VAE, etc.)



- Image Generator和分类器连接一块, 我们的目标函数就是

$$z^* = \arg \max_z y_i \quad (4)$$

找出来的图片长什么样: $X^* = G(z^*)$

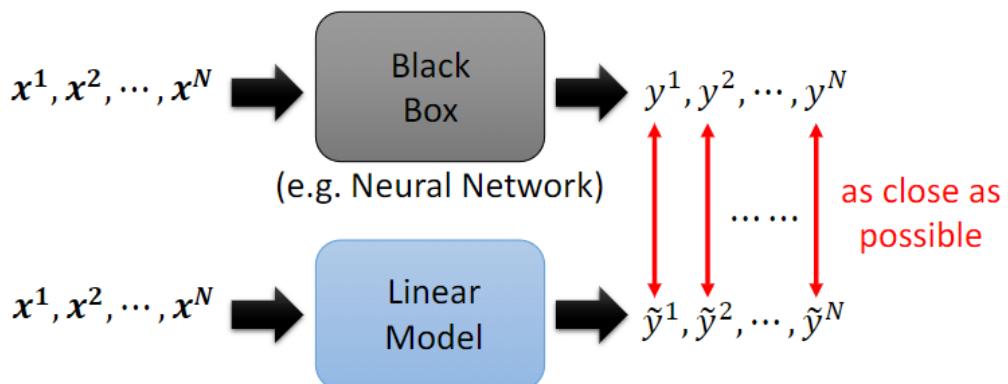


- 很work (表面上看.....), 感觉就是强行解释..... (自欺欺人)

Outlook

用一个比较简单的模型来模仿比较复杂的模型, 如果我们知道简单模型的行为, 那么也可以由此知道复杂模型的行为。

有点像同态的思想



(弹幕有提到知识蒸馏.....)

*Local Interpretable Model-Agnostic Explanations (LIME): 阅读文献以及作业