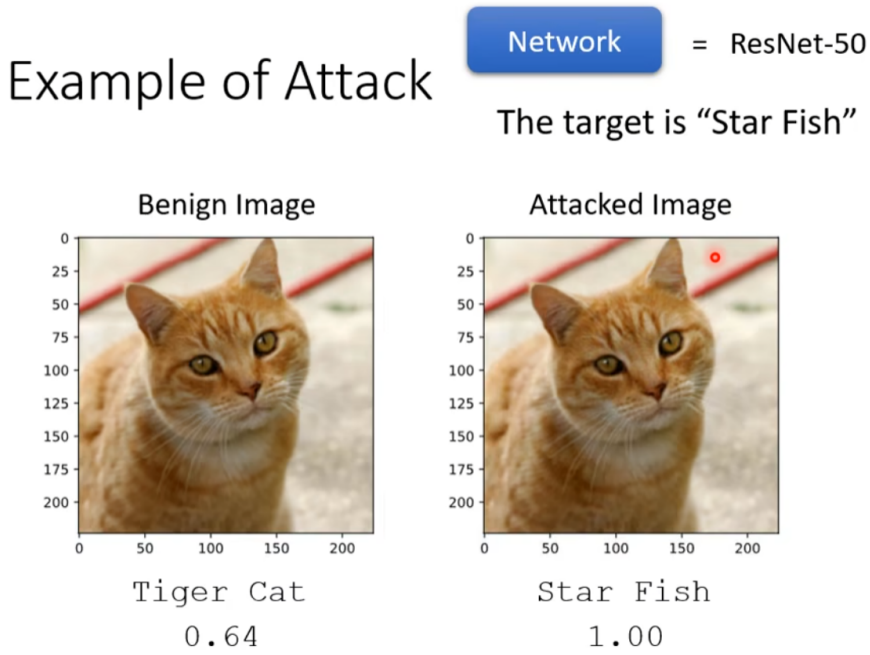
恶意攻击

动机

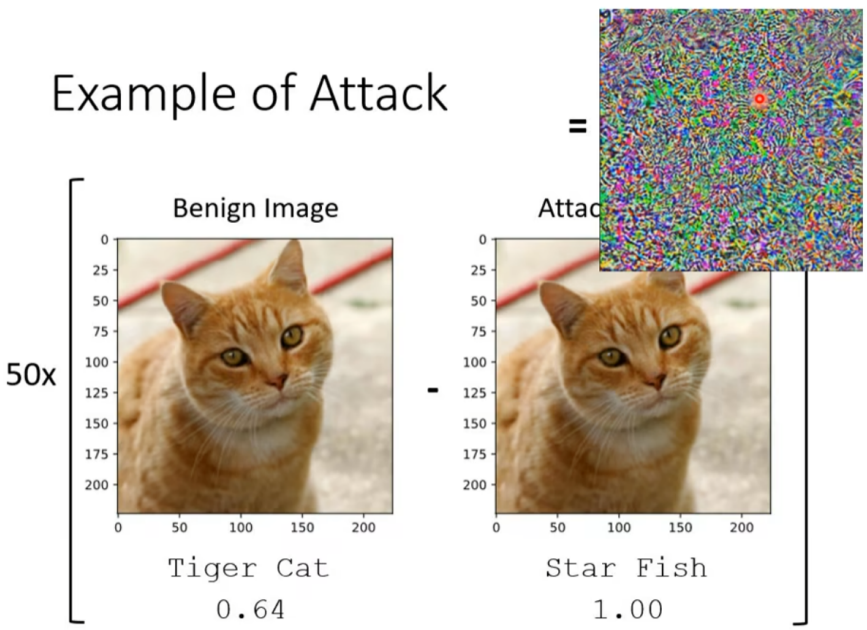
只有正确率高是不够的，还需要应付恶意行为。

常见攻击手段：通过加入微小的信息，干扰判断

- 无目的：输出错误内容
- 有目的：输出其他特定内容



加入的数据肉眼无法辨别，相减放大50倍，可以看出添加了攻击信息。

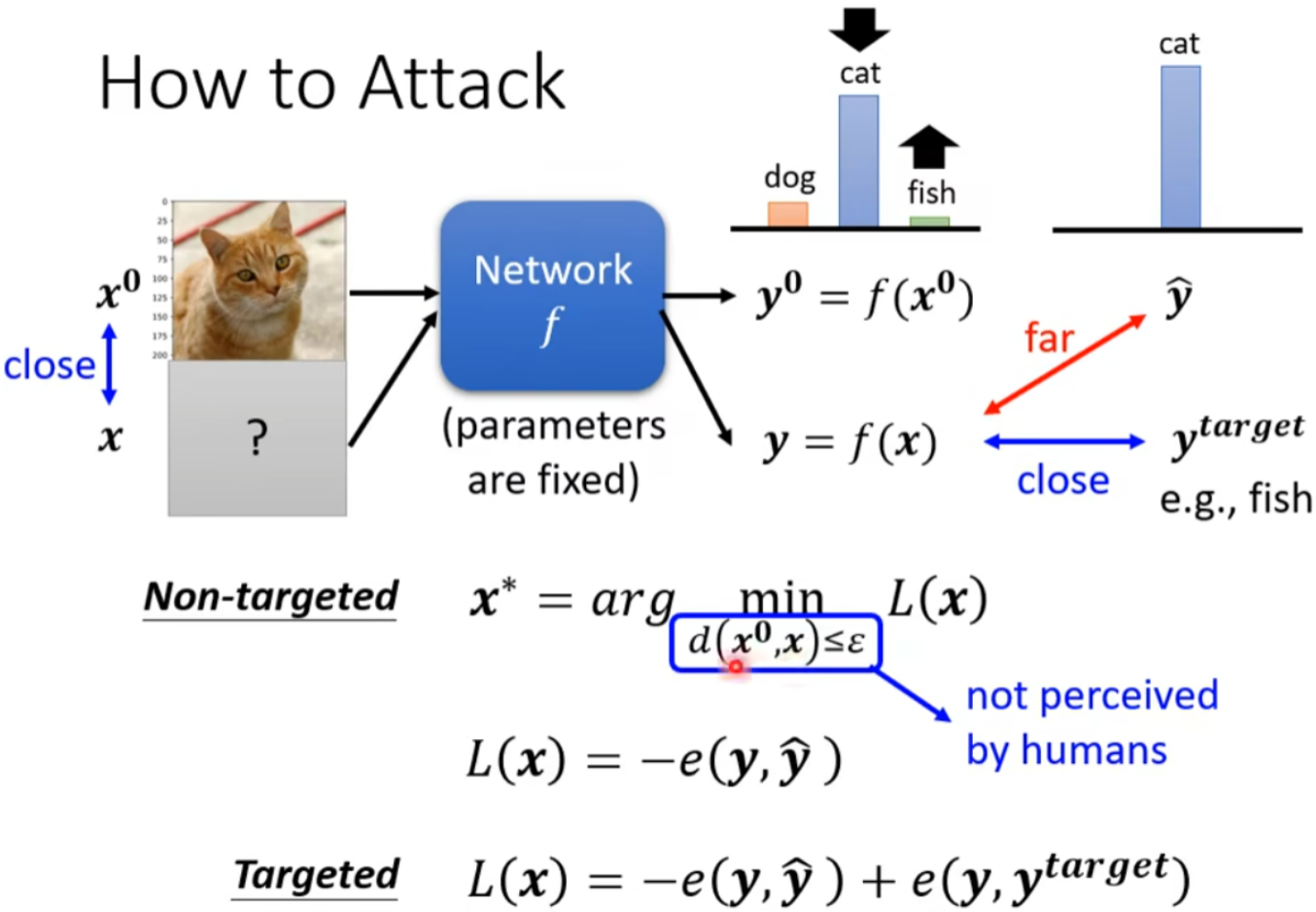


白箱攻击

知道网络参数，情况比较少（不开源）。类似于正常训练。

- 在无目标攻击中，把loss定义为负的交叉熵（只希望预测结果和原本结果越大越好）
- 在目标攻击中，一方面希望误差大，一方面又希望离目标结果接近

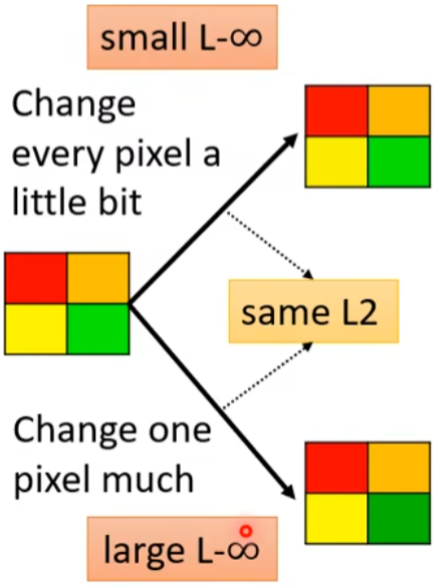
但输入的的x不是无限制的，是希望不被人眼察觉到，因此要对x限制，和真实图像的差距小。



那么如何计算真实和攻击的距离 $d(x^0, x) <= \varepsilon$

假设$\Delta x_i = x - x_i^0$，我们可以利用这些公式计算d:

- L2-norm: $d(x^0, x) = ||\Delta x||_2 = \sum_i (\Delta x_i)^2$
- L-infinity: $d(x^0, x) = ||\Delta x||_\infty = max\{|\Delta x_1|, |\Delta x_2|, |\Delta x_3|, ...\}$

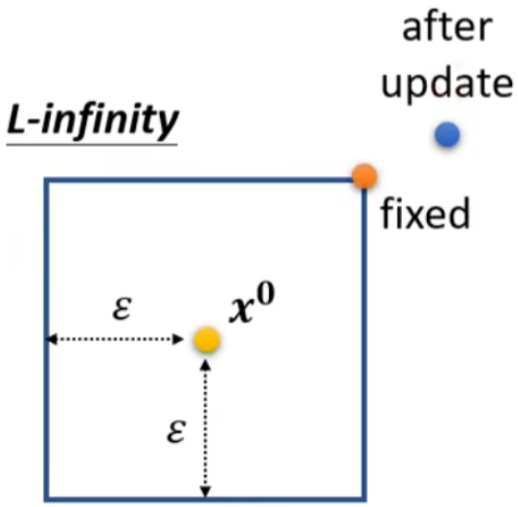


但由于要考虑是否被人类发现，变化不能集中在一点（图中右下角变化大，其他不变，L2相同，但$L-\infty$很大），这样还是可以被人眼察觉被攻击。因此要$L-\infty$尽可能小。

- 那么如何训练 $x^*=\arg min_{d(x^0,x)<=\varepsilon}L(x)$

原本机器学习训练Gradient Descent是利用$x^t \leftarrow x^{t-1}-\eta g$

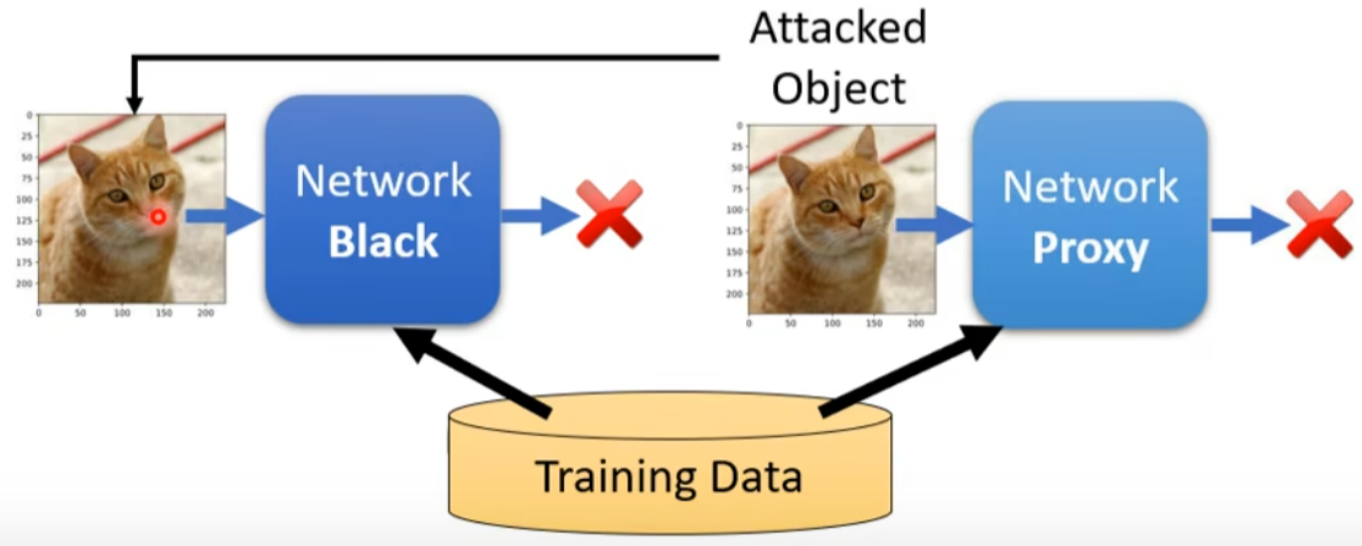
那么就多加一个限制就行：if $d(x^0,x)>\varepsilon$ then $x^t \leftarrow fix(x^t)$



黑箱模型

不知道模型参数（不知道Gradient）

- 如果知道测试集：那就自己训练代理网络，使用代理网络生成被攻击的对象



自己训练好，把模型给攻击的目标模型

- 如果不知道训练集：自己造数据集，利用输入输出，造标签数据集（和上述一样）

白箱攻击的成功率最高，黑箱攻击也有很高的成功率（当骗过多个模型的数据，也大概率能骗过目标网络）。

防御

- 主动防御

（1）对输入加入一个filter，用于轻微模糊化处理（但也可能被敌人当作是网络的一部分，反制）。

（2）对输入压缩解压缩（失真）

（3）用Generator重新生成输入

- 被动防御

（1）自己也不知道图片会咋改变（Randomization，但也可能被反制）

（2）Adversarial Training：训练的阶段就生成一些攻击的资料，不断增加数据集（也可以看作是一种**数据增强方法**），但这种手段挡不住新的攻击算法生成的攻击资料。