这一周的主要内容是如何评估我们的模型，并根据评估结果快速高效地优化我们的机器学习系统。

1. 机器学习策略必要性

如果我们的学习系统没有达到我们理想的目标（比如，训练一个猫的分类器，准确率只有90%），这时，我们可能有很多想法去优化我们的系统，比如：

- Collect more data
- Collect more diverse training data set
- Train algorithm longer with gradient descent
- Try Adam instead of gradient descent
- Try bigger network
- Try small network
- Try dropout
- Add L_2 regularization
- Modify network architecture
    - Activation functions
    - # hidden units
    - ... ...

面对这么多的工具，该如何选择才能快速高效地优化我们的模型呢？这需要一些分析策略，这些都是老师在搭建和部署大量深度学习产品时学到的经验和教训，可以帮助我们避免踩坑。

2. 正交化

正交化（Orthogonalization）或正交性是一种系统设计属性，其确保修改算法的指令或部分不会对系统的其它部分产生或传播副作用。相互独立地验证使得算法变得更简单，减少了测试和开发的时间。

在监督学习模型中，以下的4个假设需要真实且是相互正交的：

1. 系统在训练集（train set）上表现的好
    - 否则，使用更大的神经网络、更好的优化算法
2. 系统在开发集（dev set）上表现的好
    - 否则，使用正则化或加入更多的训练样本
3. 系统在测试集上表现的好
    - 否则，可以尝试使用更大的开发集进行验证
4. 系统在真实的系统环境中表现的好
    - 可能测试集没有设置正确或代价函数评估指标有误，需要改变测试集或代价函数

面对遇到的各种问题，正交化能够帮助我们更为精准有效地解决问题。

早停法（Early Stopping）是一项重要的技术，但老师不太推荐。因为早期停止，虽然可以改善开发集的拟合表现，但是对训练集的拟合就不够好。因为对两个不同的“功能”都有影响，所以早停法不具有正交性。

3. 单一数字评估指标

在训练机器学习模型的时候，无论是调整超参数，还是尝试更好的优化算法，为问题设置一个**单一数字评估指标**，可以更好更快地评估模型。

示例一

对于二分类问题，常用的评价指标是**查准率（Precision）**和**查全率（Recall）**。假设我们又A和B两个分类器，其两项指标分别如下：

Classifier	Precision	Recall	F1 Score
A	95%	90%	92.4%
B	98%	85%	91.0%

由上表可以看出，单从Precision或Recall指标出发很难比较两个分类器的优劣。

这里以Precision和Recall为基础，构成一个综合指标**F1 Score**，那么我们利用F1 Score便可以更容易判断分类器A更好。

> - Precision：预测为正类中有多少是真正的正类；
> - Recall：样本中有多少正分类被预测到了
> - $F1 = \frac{1}{\frac{1}{P}+\frac{1}{R}} = \frac{2PR}{P+R}$

示例二

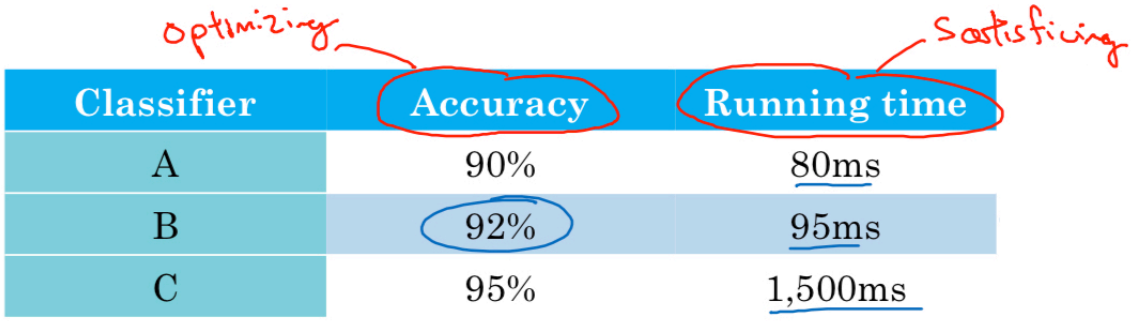
针对某一问题的多个分类器在不同国家的错误率结果：

Algorithm	US	China	India	Other	Average
A	3%	7%	5%	9%	6%
B	5%	6%	5%	10%	6.5%
C	2%	3%	4%	5%	3.5%
D	5%	8%	7%	2%	5.25%
E	4%	5%	2%	4%	3.75%
F	7%	11%	8%	12%	9.5%

模型在各个地区有不同的表现，这里用地区的平均值来对模型效果进行评估，转换为单一数字评估指标，就可以很容易的得出表现最好的模型。

4. 满足和优化指标

假设有三个分类器性能表现如下：



Classifier	Accuracy	Running time
A	90%	80ms
B	92%	95ms
C	95%	1,500ms

用时对于某一问题，对模型的效果有一定的要求，如要求模型准确率尽可能高，运行时间在100ms以内。这里把Accuracy作为了**优化指标（Optimizing metric）**，以Running time作为**满足指标（Satisfying metric）**。从而选择B作为满足条件的最优分类器。

一般地，如果要考虑N个指标，则选择一个指标为优化指标，其它N-1个指标都是满足指标。

5. 训练/开发/测试集

训练、开发、测试集选择设置的一些规则和建议：

1. 训练、开发、测试集的设置会对产品带来非常大的影响；
2. 在选择**开发集（Dev set）**和**测试集（Test set）**时要使二者来自同一分布，且从所有数据中随机选择；
3. 所选择的开发集和测试集中的数据，要与未来想要或者能够得到的数据相似；
4. 设置的测试集（Test set）只要足够大，使其能够在过拟合的系统中给出高方差的结果就可以，也许10000左右的数目足够；
5. 设置的开发集（dev set）只要足够使其能够检测不同算法、不同模型之间的优劣差异即可，百万大数据中1%的大小就足够。

6. 开发测试集大小

过去数据量较小（小于1万）时，通常将数据集按照以下比例进行划分：

- 无开发集的情况下：训练集：测试集 = 70：30
- 有开发集的情况下：训练集：开发集：测试集 = 60：20：20

这是为了保证开发集和测试集有足够的数据。

现在机器学习时代数据规模普遍较大，例如100万数据量，这时将相应比例设置为98%：1%：1%或99%：1%就已经能保证开发集和测试集的规模足够。

开发集（dev set）的大小应该设置的足够用于评估几个不同的模型；测试集（test set）的大小也因设置的足够提高系统整体性能的可信度。应根据实际情况对数据集灵活地进行划分。

7. 改变开发、测试集和评估指标

针对某一问题，我们设置开发集和评估指标后，这就像把目标定在某个位置，后面的过程就聚焦在该位置上。但有时候在这个过程中，可能发现目标位置设置错了，所以要移动改变我们的目标。

示例一

假设有两个猫的图片分类器：

- 评估指标：分类错误率（error rate）
- 算法A：3% error
- 算法B：5% error

这样看起来，算法A更好。但在实际测试过程中，算法A可能因为某些原因，将很多色情图片错误分类成了猫，这是不能接受的，所以，虽然算法A的错误率低，但是它不是一个好的算法。

这个时候我们就需要改变开发集、测试集或者评估指标。

假设原来我们的评估指标如下：

$$\text{Error} = \frac{1}{m_{dev}} \sum_{i=1}^{m_{dev}} \mathbb{I}\{y_{pred}^{(i)} \neq y^{(i)}\}$$

该指标对色情图片和非色情图片一视同仁，但是我们希望，分类器不会错误将色情图片标记为猫。

修改的方法，在其中加入权重$w^{(i)}$：

$$\text{Error} = \frac{1}{\sum w^{(i)}} \sum_{i=1}^{m_{dev}} w^{(i)} \mathbb{I}\{y_{pred}^{(i)} \neq y^{(i)}\}$$

其中：

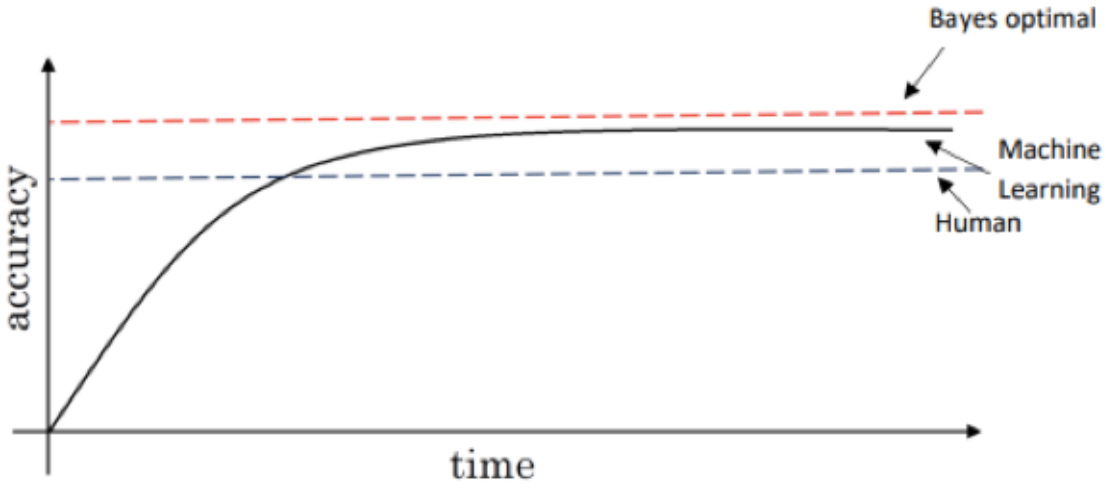
$$w^{(i)} = \begin{cases} 1 & \text{如果} x^{(i)} \text{不是色情图片} \\ 10 & \text{如果} x^{(i)} \text{是色情图片} \end{cases}$$

这样通过设置权重，当算法将色情图片分类为猫时，误差项将快速变大。

总结来说：**如果评估指标无法正确评估算法的排名，则需要重新定义一个新的评估指标。**

8. 与人类表现水平比较

很多机器学习模型的诞生是为了取代人类工作，因此其表现也会更人类表现作比较。



上图展示了随着时间的推进，机器学习系统性能的发展。一般地，当机器学习超过人的表现水平后，它的进步速度逐渐变得缓慢，最终性能无法超过某个理论上限，这个上限被称为贝叶斯最优误差（Bayes Optimal Error）。

贝叶斯最优误差一般认为是理论上可能达到的最优误差，换句话说，其就是理论最优函数，任何从 x 到 y 的函数的精确到都不可能超过这个值。例如，对于语音识别，某些音频片段嘈杂到基本不可能知道说的是什么，所以完美的识别率不可能达到100%。

因为人类对一些自然感知问题的表现水平十分接近贝叶斯最优误差，所以当机器学习系统的表现超过人类后，就没有太多继续改善的空间了。

也因此，只要建立的机器学习模型的表现还没达到人类的表现水平时，就可以通过各种手段来提升它，比如：

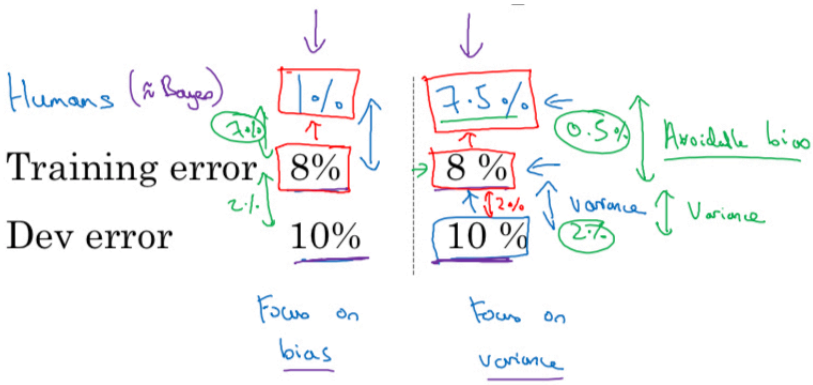
- 采用人工标记过的数据进行训练；
- 通过人工误差分析了解为什么人能够正确识别；
- 进行偏差、方差分析。

但当模型的表现超过人类后，这些手段能起的作用就微乎其微了。

9. 可避免偏差

通过与贝叶斯最优误差（Bayes Optimal Error），或者说，与人类表现水平比较，可以衡量一个机器学习模型表现，由此判断后续操作应该注重减少偏差还是减少方差。

可避免偏差（Avoidable bias）：模型在**训练集（Train set）**上的误差与人类表现水平的差值。

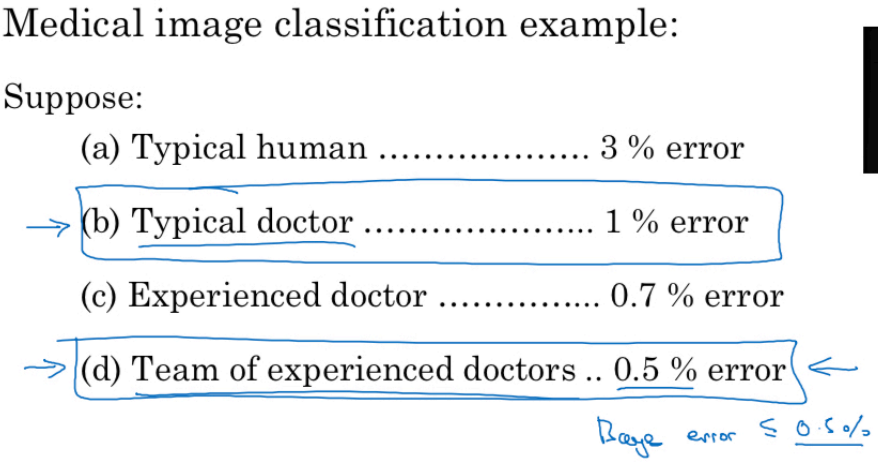


上图左边的例子，可避免偏差（Avoidable bias）与模型在开发集与训练集上错误率差值大，因此应专注于减小模型偏差；

而对于右边的例子，应专注于减小模型的方差。

10. 理解人类表现水平

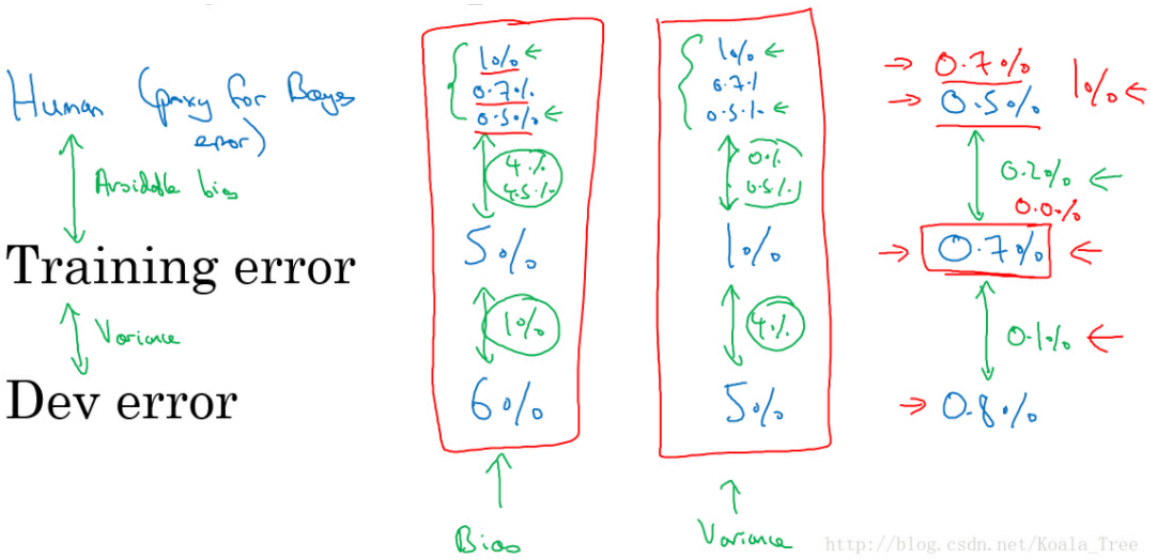
我们一般用**人类水平误差（Human-level Error）**来代表贝叶斯最优误差（或简称贝叶斯误差）。对于不同领域的例子，不同人群由于其经验水平不一，错误率也不一样，该如何确定人类水平误差呢?



一般来说，我们将表现最好的作为人类水平误差。在减小误诊了的背景下，人类水平误差在这种情形下应定义为：0.5% error。

如果是为了部署系统或者做研究分析的背景下，也许超过一名普通医生即可，即人类水平误差在这种情形下应定义为：1% error。

人类水平误差基准的不同选择，对模型的优化方向有影响，如下图所示：



当机器学习模型的表现超过了人类水平误差时，很难再通过人的直觉去判断模型还能够往什么方向优化以提高性能。

11. 超过人的表现

假设在一个分类任务中：

	Classification error (%)	
	Scenario A	Scenario B
Team of humans	0.5	0.5
One human	1.0	1
Training error	0.6	0.3
Development error	0.8	0.4

small than human-level error

- Scenario A

  用人类表现水平估计贝叶斯误差为0.5%，可避免偏差（Avoidable bias）为0.1%，小于(dev error - train error)，应采取减小方差的技术。

- Scenario B

  模型误差小于人类表现水平，此时很难判断该去减小偏差还是减小方差。

现实中很多问题，机器学习模型的表现能显著超过人类表现水平，尤其是在结构化的数据上，比如：

- 在线广告
- 产品推荐

- 贷款审核

12. 提升模型表现

基本假设：

1. 模型在训练集上有很好的表现；
2. 模型在开发/测试集上表现也很好。