Cross_Validation 交叉验证

1. leave-one-out cross-validation， LOOCV方法

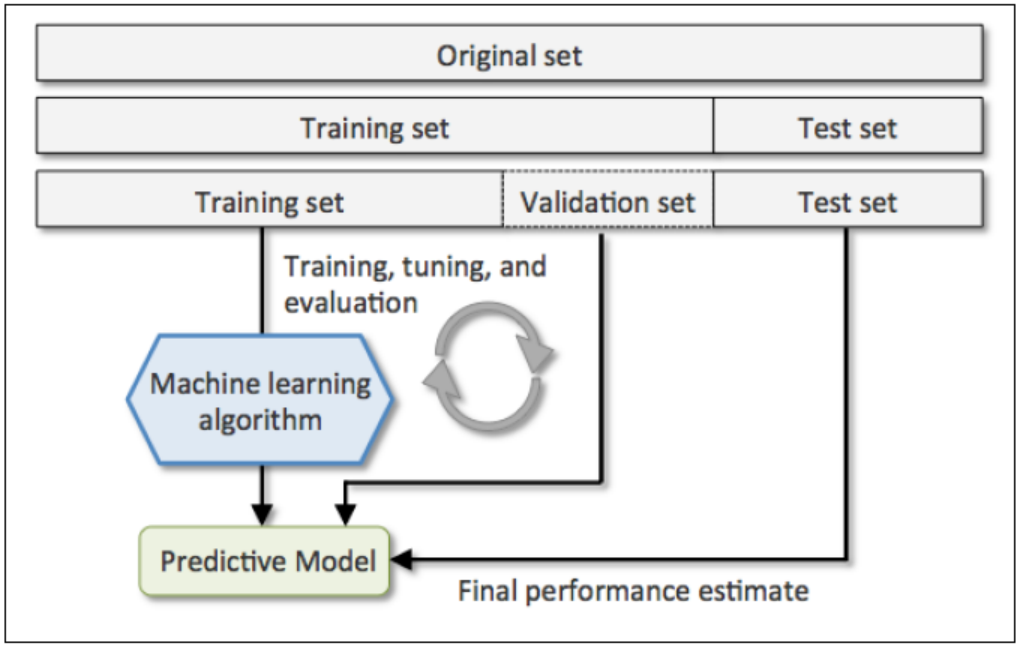
像Test set approach一样，LOOCV方法也包含将数据集分为训练集和测试集这一步骤。但是不同的是，我们现在只用一个数据作为测试集，其他的数据都作为训练集，并将此步骤重复N次（N为数据集的数据数量）。

假设我们现在有n个数据组成的数据集，那么LOOCV的方法就是每次取出一个数据作为测试集的唯一元素，而其他n-1个数据都作为训练集用于训练模型和调参。结果就是我们最终训练了n个模型，每次都能得到一个MSE。而计算最终test MSE则就是将这n个MSE取平均。

2. holdout cross validation 留出法

在机器学习任务中，拿到数据后，我们首先会将原始数据集分为三部分：**训练集、验证集和测试集。**

训练集用于**训练模型**，验证集用于**模型的参数选择配置**，测试集对于模型来说是未知数据，用于**评估模型的泛化能力**。



这个方法操作简单，只需随机把原始数据分为三组即可。

不过如果只做一次分割，它对训练集、验证集和测试集的样本数**比例**，还有分割后数据的分布是否和原始数据集的**分布**相同等因素比较敏感，不同的划分会得到不同的最优模型，而且分成三个集合后，用于训练的数据**更少**了。

3. K-fold Cross Validation, K折交叉验证

另外一种折中的办法叫做K折交叉验证，和LOOCV的不同在于，我们每次的测试集将不再只包含一个数据，而是多个，具体数目将根据K的选取决定。比如，如果K=5，那么我们利用五折交叉验证的步骤就是：

- 第一步，不重复抽样将原始数据随机分为 k 份。
- 第二步，每一次挑选其中 1 份作为测试集，剩余 k-1 份作为训练集用于模型训练。
- 第三步，重复第二步 k 次，这样每个子集都有一次机会作为测试集，其余机会作为训练集。
- 在每个训练集上训练后得到一个模型，
- 用这个模型在相应的测试集上测试，计算并保存模型的评估指标，
- 第四步，计算 k 组测试结果的平均值作为模型精度的估计，并作为当前 k 折交叉验证下模型的性能指标。

K越大，每次投入的训练集的数据越多，模型的Bias越小。但是K越大，又意味着每一次选取的训练集之前的相关性越大（考虑最极端的例子，当k=N，也就是在LOOCV里，每次都训练数据几乎是一样的）。而这种大相关性会导致最终的test error具有更大的Variance。
一般来说，根据经验我们一般选择k=5或10。

此外：

4. 多次 k 折交叉验证再求均值，例如：10 次 10 折交叉验证，以求更精确一点。
5. 划分时有多种方法，例如对非平衡数据可以用分层采样，就是在每一份子集中都保持和原始数据集相同的类别比例。
6. 模型训练过程的所有步骤，包括模型选择，特征选择等都是在单个折叠 fold 中独立执行的。

reference:
https://zhuanlan.zhihu.com/p/24825503?utm_source=tuicool&utm_medium=referral