

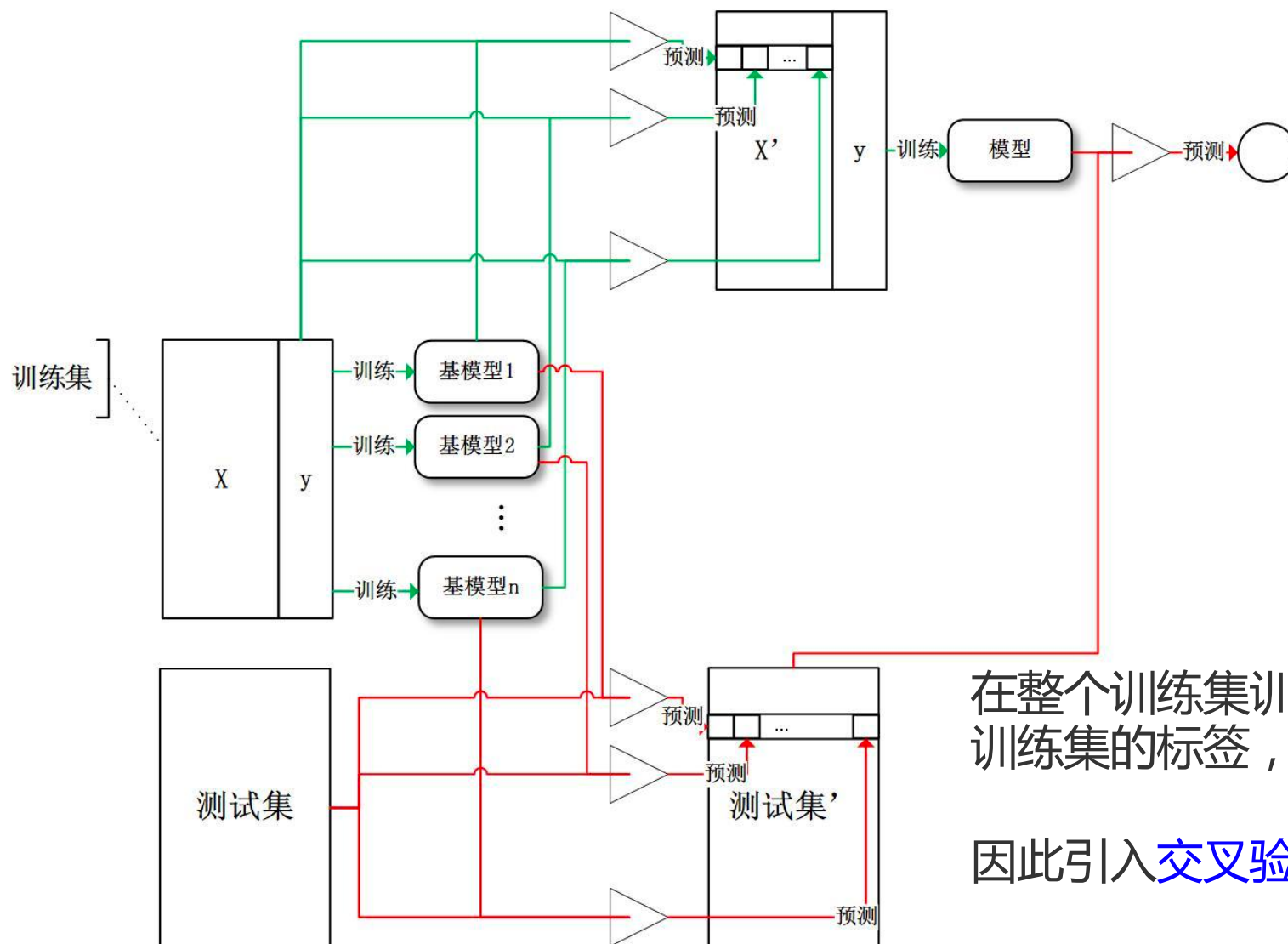
模型融合

卿来云

► 堆栈泛化 (Stacking)

- Stacking模型本质上是一种分层的结构，由Wolpert在1992年提出
- 二级Stacking：
 - 将训练好的基模型对训练集进行预测
 - 新的训练集：第 j 个基模型对第 i 个训练样本的预测值将作为新的训练集中第 i 个样本的第 j 个特征值
 - 新的测试集：所有基模型的对测试集的预测
 - 在新的训练集上训练模型，在新的测试集上进行预测

stacking

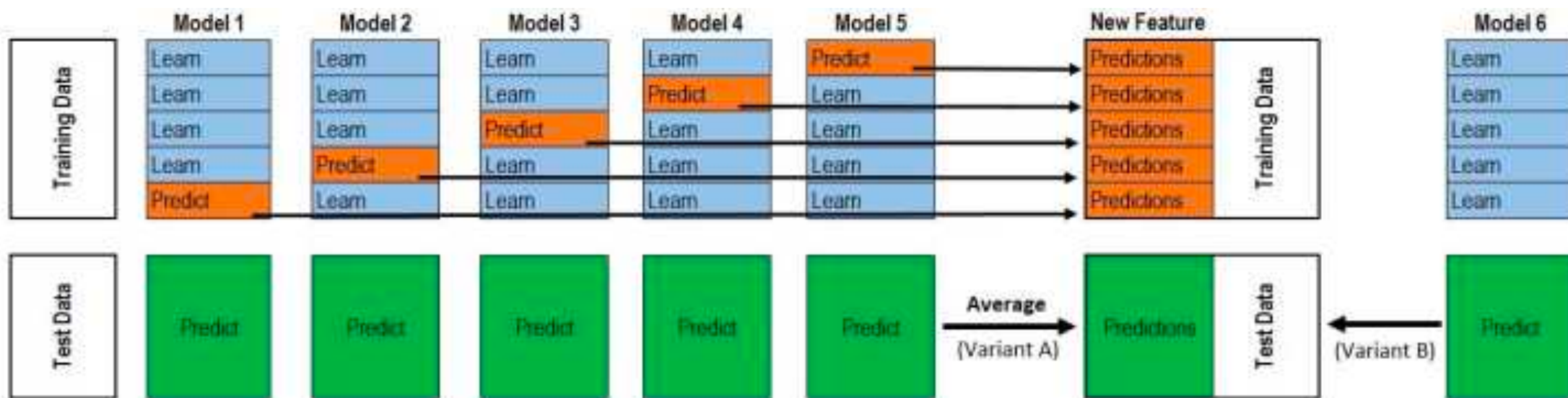



在整个训练集训练的模型反过来去预测训练集的标签，过拟合会非常非常严重

因此引入交叉验证→ blending

► 交叉融合 (Blending)

- Blending是由Netflix获胜者提出来的一个词，与堆栈泛化很像，但更简单且信息泄露的风险更低。





```
def get_oof(clf, x_train, y_train, x_test):
    oof_train = np.zeros((ntrain,))
    oof_test = np.zeros((ntest,))

    #NFOLDS行, ntest列的二维array
    oof_test_skf = np.empty((NFOLDS, ntest))

    #循环NFOLDS次
    for i, (train_index, test_index) in enumerate(kf):
        x_tr = x_train[train_index]
        y_tr = y_train[train_index]
        x_te = x_train[test_index]
        clf.fit(x_tr, y_tr)

        oof_train[test_index] = clf.predict(x_te)
        #固定行填充, 循环一次, 填充一行
        oof_test_skf[i, :] = clf.predict(x_test)

    #axis=0,按列求平均, 最后保留一行
    oof_test[:] = oof_test_skf.mean(axis=0)

    #转置, 从一行变为一列
    return oof_train.reshape(-1, 1), oof_test.reshape(-1, 1)
```

► 案例分析

- Titanic
 - <https://www.kaggle.com/arthurtok/introduction-to-ensembling-stacking-in-python>
- Blending.py

THANK YOU



AI100