参考文献来源[1]<https://blog.csdn.net/MRxjh/article/details/78499801>

[2] <https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html>

## (sklearn)逻辑回归linear\_model.LogisticRegression用法

class sklearn.linear\_model.LogisticRegression(penalty=’l2’, dual=False, tol=0.0001, C=1.0, fit\_intercept=True, intercept\_scaling=1, class\_weight=None, random\_state=None, solver=’liblinear’, max\_iter=100, multi\_class=’ovr’, verbose=0, warm\_start=False, n\_jobs=1)

### 参数：

penalty: 一个字符串，制定了正则化策略。

* 如果为'l2'，则优化目标函数为：为极大似然函数。
* 如果为'l1'，则优化目标函数为：为极大似然函数。

dual：一个布尔值。如果为True，则求解对偶形式（只是在penalty='l2' 且solver='liblinear' 有对偶形式）；如果为False，则求解原始形式。

C：一个浮点数，它指定了惩罚系数的倒数。如果它的值越小，则正则化越大。

fit\_intercept：一个布尔值，制定是否需要b 值。如果为False，则不会计算b值（模型会假设你的数据已经中心化）。

intercept\_scaling：一个浮点数，只有当 solver='liblinear' 才有意义。当采用 fit\_intercept 时，相当于人造一个特征出来，该特征恒为 1，其权重为b。在计算正则化项的时候，该人造特征也被考虑了。因此为了降低人造特征的影响，需要提供 intercept\_scaling。

class\_weight：一个字典或者字符串'balanced'。

* 如果为字典：则字典给出了每个分类的权重，如{class\_label:weight}。
* 如果为字符串 'balanced'：则每个分类的权重与该分类在样品中出现的频率成反比。
* 如果未指定，则每个分类的权重都为 1。

max\_iter：一个整数，指定最大迭代数。

random\_state：一个整数或者一个RandomState实例，或者None。

* 如果为整数，则它指定了随机数生成器的种子。
* 如果为RandomState实例，则指定了随机数生成器。
* 如果为None，则使用默认的随机数生成器。

solver：一个字符串，指定了求解最优化问题的算法，可以为如下的值。

* 'newton-cg'：使用牛顿法。
* 'lbfgs'：使用L-BFGS拟牛顿法。
* 'liblinear' ：使用 liblinear。
* 'sag'：使用 Stochastic Average Gradient descent 算法。

**注意：**

对于规模小的数据集，'liblearner'比较适用；对于规模大的数据集，'sag'比较适用。

'newton-cg'、'lbfgs'、'sag' 只处理penalty=‘12’的情况。

tol：一个浮点数，指定判断迭代收敛与否的一个阈值。

multi\_class：一个字符串，指定对于多分类问题的策略，可以为如下的值。

* 'ovr' ：采用 one-vs-rest 策略。
* 'multinomial'：直接采用多分类逻辑回归策略。

verbose：一个正数。用于开启/关闭迭代中间输出的日志。

warm\_start：一个布尔值。如果为True，那么使用前一次训练结果继续训练，否则从头开始训练。

n\_jobs：一个正数。指定任务并行时的 CPU 数量。如果为 -1 则使用所有了用的 CPU。

### 返回值

coef\_：权重向量。

intercept：b值。

n\_iter\_：实际迭代次数。

### 方法

fix(X,y[,sample\_weight])： 训练模型。

predict(X)：用模型进行预测，返回预测值。

score(X,y[,sample\_weight])：返回（X，y）上的预测准确率（accuracy）。

predict\_log\_proba（X）：返回一个数组，数组的元素一次是 X 预测为各个类别的概率的对数值。

predict\_proba（X）： 返回一个数组，数组元素一次是 X 预测为各个类别的概率的概率值。

sparsify()：将系数矩阵转换为稀疏格式。

set\_params（\*\* params）：设置此估计器的参数。

decision\_function（X）：预测样本的置信度分数。

densify()：将系数矩阵转换为密集阵列格式。