算法思想

> 所有未作说明的类变量即无关变量

KNN

- 类变量说明
  - data：训练集数据，存储在类中
  - label：训练集标签，存储在类中
  - les：Label Encoder的一个list，存储的是每一列对应的Label Encoder
  - K：KNN对应的K值，决定判定时选取的近邻点数目
  - result：计算得到的F1 score
  - cut： 是否把G1，G2两列去除
- 类方法说明
  - prepocess：对数据进行处理，即把对应的数据集中的字符编码成整数
  - fit：根据训练集数据和标签训练模型——对于KNN而且就是简单地存储数据和标签即可
  - predict：根据测试集数据和标签进行训练，并计算得到F1 score
- 核心思想
  - 在预测一个数据点的标签时，计算它与其他所有点的距离，找到最近的k个，然后统计这个K个的标签，某种标签的数量大于50%时，即判定它为对应的相同标签（及格或者不及格）

SVM

- 类变量说明
  - data：训练集数据，存储在类中
  - label：训练集标签，存储在类中
  - les：Label Encoder的一个list，存储的是每一列对应的Label Encoder
  - C：SVM对应的C值，即软间隔容忍限度，影响到SVM迭代时的计算
  - kernel：采取的核函数，实现了线性核与高斯核（rbf）
    - 使用高斯核是因为高斯核的效果相比线性核好很多，而且在SMO迭代时，采用高斯核可以得到一个大小适度的Kernel Matrix，这样的话便可以使bias的值适中，不会出现使用线性核时预测全为一个值的窘况
  - sv：训练得到的支持向量
  - sl：训练得到的支持向量对应的标签
  - alphas：支持向量对应的常数系数，用于分类
  - weight：训练得到的权重list
  - bias：训练得到的偏差
  - sigma：高斯核对应的sigma参数，用于改进高斯核效果
  - result：计算得到的F1 score
  - cut： 是否把G1，G2两列去除
- 类方法说明（未作说明的方法未内部方法，不需要外部调用）
  - prepocess：对数据进行处理，即把对应的数据集中的字符编码成整数
  - fit：根据训练集数据和标签训练模型

    - predict：根据测试集数据和标签进行训练，并计算得到F1 score
- 内部类说明
    - KernelFunction：提供线性核与高斯核
    - SMO：用于SVM的支持向量求解
- 核心思想
    - 每次迭代中，先找到一个违背KKT原则的变量（本来应该寻找违背程度最大的变量，但此处只寻找了任意违背KKT的变量）
    - 随机选取另一个与上述变量不同的变量
    - 固定其他变量，通过SMO算法中的关系式得到alpha[i]与alpha[j]的迭代值
    - 通过上述计算的结果求解新的bias
    - 不断迭代直到没有大改变为止（设定了一个小常量，小于这个值的变化将被忽略，用于加速迭代过程）

Logistic Regression

- 类变量说明
    - data：训练集数据，存储在类中
    - label：训练集标签，存储在类中
    - les：Label Encoder的一个list，存储的是每一列对应的Label Encoder
    - weight：模型的参数，提供一个映射作用，在训练中不断迭代
    - alpha：每次步进长度
    - iteration：迭代次数，迭代这么多次后结束迭代
    - result：计算得到的F1 score
    - cut： 是否把G1，G2两列去除
- 类方法说明
    - prepocess：对数据进行处理，即把对应的数据集中的字符编码成整数
    - fit：根据训练集数据和标签训练模型——对于KNN而且就是简单地存储数据和标签即可
    - predict：根据测试集数据和标签进行训练，并计算得到F1 score
- 核心思想
    - 初始化weight参数
    - 开始迭代
    - 获取下降梯度gradient
    - weight下降步进长度alpha与梯度gradient的乘积
    - 迭代iteration次

实验结果

KNN

- K=10
    - 不采用G1，G2

      ```
      F1 score of KNN is 0.9152542372881356
      ```

    - 采用G1，G2

      ```
      F1 score of KNN is 0.8734177215189874
      ```

- K=15
    - 不采用G1，G2

      ```
      F1 score of KNN is 0.9249999999999999
      ```

    - 采用G1，G2

```
F1 score of KNN is 0.8794326241134752
```

- K=20
    - 不采用G1，G2

```
F1 score of KNN is 0.9012345679012346
```

    - 采用G1，G2

```
F1 score of KNN is 0.9171974522292994
```

SVM

- 不采用G1，G2
    - C=5.，采用高斯核，sigma=1，得到的F1 score为0.7445255474452555
    - C=5.，采用高斯核，sigma=3，得到的F1 score为0.6857142857142857
    - C=5.，采用高斯核，sigma=5，得到的F1 score为0.7014925373134329
    - C=1.，采用高斯核，sigma=1，得到的F1 score为0.7299270072992702
    - C=5.，采用线性核，因为全预测的是不及格，因此没有F1 score（TP + FP = 0，分母为0），只得到预测准确率TN / (TN + FN)为0.3157894736842105
    - C=1.，采用线性核，因为全预测的是不及格，因此没有F1 score（TP + FP = 0，分母为0），只得到预测准确率TN / (TN + FN)为0.368
- 采用G1，G2
    - C=5.，采用高斯核，sigma=1，得到的F1 score为0.6666666666666666
    - C=5.，采用高斯核，sigma=3，得到的F1 score为0.726027397260274
    - C=5.，采用高斯核，sigma=5，得到的F1 score为0.7132867132867133

Logistic Regression

- 步长0.0001，迭代次数100000
    - `F1 score of logistic regression is 0.95705521472392`64
- 步长0.01，迭代次数100000
    - `F1 score of logistic regression is 0.93243243243243`23
- 步长0.0001，迭代次数1000
    - `F1 score of logistic regression is 0.83695652173913`04