**Ex3 Report**

自硕21 崔晏菲 2021210976

**1. 参数解释**

(1) C：这是一个惩罚参数。C越大，相当于惩罚松弛变量，希望松弛变量接近0，即对误分类的惩罚增大，趋向于对训练集全分对的情况，这样对训练集测试时准确率很高，但容易造成过拟合，泛化能力弱。C越小，对误分类的惩罚减小，允许容错，将他们当成噪声点，泛化能力较强。

(2) kernel：这是核函数。可选择”linear”, “poly”, “sigmoid”, “precomputed”, “rbf”等。

“linear”：线性核，即；

“poly”：多项式核，即；

“rbf”：高斯核，即；

“sigmoid”：其实是双曲正切核，即；

“precomputed”：自己提前计算好核函数矩阵，这时候算法内部就不再用核函数去计算核矩阵，而是直接用你给的核矩阵，核矩阵需要为的。

(3) degree：当核函数选择”poly”时，所需要的degree参数。

(4) gamma：即”poly”, “rbf”, “sigmoid”核中的，这个参数有两个选项：”scale”, “auto”。

“scale”：；

“auto”：。

(5) coef0: 截距项，即”poly”, “sigmoid”中的。

(6) shrinking：布尔值，是否采用启发式收缩方式，bool类型，可选参数，默认为True。

(7) probability：布尔值，是否启用概率估计，可选参数，默认为False，这必须在调用fit()之前启用，并且会fit()方法速度变慢。

(8) tol：svm停止训练的误差精度，float类型，可选参数，默认为。

(9) class\_weight：类别权重，dict类型或str类型，可选参数，默认为None。给每个类别分别设置不同的惩罚参数C，如果没有给，则会给所有类别都给C=1，即前面参数指出的参数C。如果给定参数’balance’，则使用y的值自动调整与输入数据中的类频率成反比的权重，这是为了解决正负样本不平衡的问题。

(10) max\_iter：最大迭代次数，int类型，默认为-1，表示不限制。

(11) random\_state：数据洗牌时的种子值，int类型，可选参数，默认为None。伪随机数发生器的种子,在混洗数据时用于概率估计。

对不同的任务，应当选择不同的参数。当数据在欧式空间中线性可分时，我们通常使用线性核，若在欧式空间中存在非线性，我们会通常用多项式核来学习。若样本数据实在概率空间中的，我们通常会选择高斯核。一般来说，对于特定的任务，我们会通常试验多个不同的核函数。

**2. 实验**

我在TrainingSet-1上进行了5折交叉验证，并且对每折都在TestSet-1上进行了测试。

**2.1 线性核**

线性核函数选取的三组参数和结果如下：

kernel = "linear", C = 0

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fold** | **train accuracy** | **train sensi** | **train speci** | **valid accuracy** | **valid sensi** | **valid speci** | **test accuracy** | **test sensi** | **test speci** |
| 1 | 0.510000 | 1.000000 | 0.000000 | 0.510000 | 1.000000 | 0.000000 | 0.498633 | 1.000000 | 0.000000 |
| 2 | 0.510000 | 1.000000 | 0.000000 | 0.510000 | 1.000000 | 0.000000 | 0.498633 | 1.000000 | 0.000000 |
| 3 | 0.510000 | 1.000000 | 0.000000 | 0.510000 | 1.000000 | 0.000000 | 0.498633 | 1.000000 | 0.000000 |
| 4 | 0.510000 | 1.000000 | 0.000000 | 0.510000 | 1.000000 | 0.000000 | 0.498633 | 1.000000 | 0.000000 |
| 5 | 0.510000 | 1.000000 | 0.000000 | 0.510000 | 1.000000 | 0.000000 | 0.498633 | 1.000000 | 0.000000 |
| Average | 0.510000 | 1.000000 | 0.000000 | 0.510000 | 1.000000 | 0.000000 | 0.498633 | 1.000000 | 0.000000 |

kernel = "linear", C = 1

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fold** | **train accuracy** | **train sensi** | **train speci** | **valid accuracy** | **valid sensi** | **valid speci** | **test accuracy** | **test sensi** | **test speci** |
| 1 | 0.800000 | 0.802451 | 0.797449 | 0.789000 | 0.778431 | 0.800000 | 0.784868 | 0.767824 | 0.801818 |
| 2 | 0.796000 | 0.791176 | 0.801020 | 0.805000 | 0.805882 | 0.804082 | 0.781222 | 0.765996 | 0.796364 |
| 3 | 0.801750 | 0.799020 | 0.804592 | 0.787000 | 0.790196 | 0.783673 | 0.793984 | 0.786106 | 0.801818 |
| 4 | 0.804250 | 0.797059 | 0.811735 | 0.769000 | 0.750980 | 0.787755 | 0.783045 | 0.767824 | 0.798182 |
| 5 | 0.806000 | 0.797059 | 0.815306 | 0.762000 | 0.764706 | 0.759184 | 0.783045 | 0.765996 | 0.800000 |
| Average | 0.801600 | 0.797353 | 0.806020 | 0.782400 | 0.778039 | 0.786939 | 0.785232 | 0.770750 | 0.799636 |

kernel = "linear", C = 10

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fold** | **train accuracy** | **train sensi** | **train speci** | **valid accuracy** | **valid sensi** | **valid speci** | **test accuracy** | **test sensi** | **test speci** |
| 1 | 0.799500 | 0.800980 | 0.797959 | 0.789000 | 0.778431 | 0.800000 | 0.786691 | 0.773309 | 0.800000 |
| 2 | 0.797250 | 0.793137 | 0.801531 | 0.805000 | 0.805882 | 0.804082 | 0.780310 | 0.764168 | 0.796364 |
| 3 | 0.801500 | 0.797549 | 0.805612 | 0.786000 | 0.786275 | 0.785714 | 0.794895 | 0.787934 | 0.801818 |
| 4 | 0.804750 | 0.798039 | 0.811735 | 0.772000 | 0.752941 | 0.791837 | 0.782133 | 0.767824 | 0.796364 |
| 5 | 0.805750 | 0.797059 | 0.814796 | 0.762000 | 0.764706 | 0.759184 | 0.783045 | 0.765996 | 0.800000 |
| Average | 0.801750 | 0.797353 | 0.806327 | 0.782800 | 0.777647 | 0.788163 | 0.785415 | 0.771846 | 0.798909 |

可见，当惩罚参数过小时，SVM无法进行分类，倾向于把所有的数据分为同一类。当C过大时，训练速度明显降低，理论上容易发生过拟合，但实际上因为数据本身是线性不可分的，而我们用的是线性核，因此没有发生过拟合。

**2.2 高斯核**

高斯核函数选取的三组参数和结果如下：

kernel = "rbf", gamma = "scale", C = 1

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fold** | **train accuracy** | **train sensi** | **train speci** | **valid accuracy** | **valid sensi** | **valid speci** | **test accuracy** | **test sensi** | **test speci** |
| 1 | 0.881750 | 0.887255 | 0.876020 | 0.794000 | 0.792157 | 0.795918 | 0.791249 | 0.793419 | 0.789091 |
| 2 | 0.882750 | 0.880882 | 0.884694 | 0.803000 | 0.813725 | 0.791837 | 0.782133 | 0.784278 | 0.780000 |
| 3 | 0.878000 | 0.872549 | 0.883673 | 0.800000 | 0.805882 | 0.793878 | 0.784868 | 0.780622 | 0.789091 |
| 4 | 0.883250 | 0.876961 | 0.889796 | 0.774000 | 0.774510 | 0.773469 | 0.790337 | 0.786106 | 0.794545 |
| 5 | 0.886000 | 0.886275 | 0.885714 | 0.784000 | 0.803922 | 0.763265 | 0.790337 | 0.797075 | 0.783636 |
| Average | 0.882350 | 0.880784 | 0.883980 | 0.791000 | 0.798039 | 0.783673 | 0.787785 | 0.788300 | 0.787273 |

kernel = "rbf", gamma = "auto", C = 1

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fold** | **train accuracy** | **train sensi** | **train speci** | **valid accuracy** | **valid sensi** | **valid speci** | **test accuracy** | **test sensi** | **test speci** |
| 1 | 0.882000 | 0.887255 | 0.876531 | 0.794000 | 0.792157 | 0.795918 | 0.791249 | 0.793419 | 0.789091 |
| 2 | 0.883250 | 0.881863 | 0.884694 | 0.804000 | 0.815686 | 0.791837 | 0.782133 | 0.784278 | 0.780000 |
| 3 | 0.878000 | 0.872549 | 0.883673 | 0.800000 | 0.805882 | 0.793878 | 0.784868 | 0.780622 | 0.789091 |
| 4 | 0.883000 | 0.876961 | 0.889286 | 0.774000 | 0.774510 | 0.773469 | 0.790337 | 0.786106 | 0.794545 |
| 5 | 0.886250 | 0.886275 | 0.886224 | 0.784000 | 0.803922 | 0.763265 | 0.790337 | 0.797075 | 0.783636 |
| Average | 0.882500 | 0.880980 | 0.884082 | 0.791200 | 0.798431 | 0.783673 | 0.787785 | 0.788300 | 0.787273 |

kernel = "rbf", gamma = "scale", C = 10

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fold** | **train accuracy** | **train sensi** | **train speci** | **valid accuracy** | **valid sensi** | **valid speci** | **test accuracy** | **test sensi** | **test speci** |
| 1 | 0.992250 | 0.988235 | 0.996429 | 0.779000 | 0.780392 | 0.777551 | 0.770283 | 0.765996 | 0.774545 |
| 2 | 0.993750 | 0.989706 | 0.997959 | 0.773000 | 0.786275 | 0.759184 | 0.769371 | 0.780622 | 0.758182 |
| 3 | 0.994500 | 0.991667 | 0.997449 | 0.772000 | 0.762745 | 0.781633 | 0.752051 | 0.745887 | 0.758182 |
| 4 | 0.993750 | 0.991176 | 0.996429 | 0.753000 | 0.768627 | 0.736735 | 0.770283 | 0.769653 | 0.770909 |
| 5 | 0.992500 | 0.989216 | 0.995918 | 0.753000 | 0.754902 | 0.751020 | 0.778487 | 0.784278 | 0.772727 |
| Average | 0.993350 | 0.990000 | 0.996837 | 0.766000 | 0.770588 | 0.761224 | 0.768095 | 0.769287 | 0.766909 |

可见，高斯核函数很好的解决了样本线性不可分的问题。通过对比gamma的取值，我们发现gamma的取值对实验结果影响不大。但是比较C的取值我们发现，当C取值较大时，样本在训练集上接近全队，但在验证集和测试集上的表现反而不如C较小的时候，这说明C值增大导致了过拟合，导致泛化能力变差。

**2.3 多项式核**

多项式核函数选取的四组参数和结果如下：

kernel = "poly", coef0 = 0, degree = 2

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fold** | **train accuracy** | **train sensi** | **train speci** | **valid accuracy** | **valid sensi** | **valid speci** | **test accuracy** | **test sensi** | **test speci** |
| 1 | 0.848500 | 0.832843 | 0.864796 | 0.722000 | 0.664706 | 0.781633 | 0.738377 | 0.687386 | 0.789091 |
| 2 | 0.844500 | 0.817647 | 0.872449 | 0.721000 | 0.652941 | 0.791837 | 0.735643 | 0.689214 | 0.781818 |
| 3 | 0.845250 | 0.811765 | 0.880102 | 0.736000 | 0.694118 | 0.779592 | 0.723792 | 0.658135 | 0.789091 |
| 4 | 0.842250 | 0.807843 | 0.878061 | 0.743000 | 0.719608 | 0.767347 | 0.723792 | 0.672761 | 0.774545 |
| 5 | 0.846250 | 0.820588 | 0.872959 | 0.719000 | 0.676471 | 0.763265 | 0.737466 | 0.698355 | 0.776364 |
| Average | 0.845350 | 0.818137 | 0.873673 | 0.728200 | 0.681569 | 0.776735 | 0.731814 | 0.681170 | 0.782182 |

kernel = "poly", coef0 = 0, degree = 3

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fold** | **train accuracy** | **train sensi** | **train speci** | **valid accuracy** | **valid sensi** | **valid speci** | **test accuracy** | **test sensi** | **test speci** |
| 1 | 0.904500 | 0.868627 | 0.941837 | 0.769000 | 0.680392 | 0.861224 | 0.776664 | 0.716636 | 0.836364 |
| 2 | 0.903250 | 0.859314 | 0.948980 | 0.765000 | 0.701961 | 0.830612 | 0.781222 | 0.707495 | 0.854545 |
| 3 | 0.901750 | 0.851961 | 0.953571 | 0.779000 | 0.705882 | 0.855102 | 0.778487 | 0.692870 | 0.863636 |
| 4 | 0.903500 | 0.852451 | 0.956633 | 0.769000 | 0.711765 | 0.828571 | 0.779398 | 0.709324 | 0.849091 |
| 5 | 0.906000 | 0.858824 | 0.955102 | 0.773000 | 0.719608 | 0.828571 | 0.781222 | 0.707495 | 0.854545 |
| Average | 0.903800 | 0.858235 | 0.951224 | 0.771000 | 0.703922 | 0.840816 | 0.779398 | 0.706764 | 0.851636 |

kernel = "poly", coef0 = 1, degree = 3

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fold** | **train accuracy** | **train sensi** | **train speci** | **valid accuracy** | **valid sensi** | **valid speci** | **test accuracy** | **test sensi** | **test speci** |
| 1 | 0.949500 | 0.944118 | 0.955102 | 0.782000 | 0.756863 | 0.808163 | 0.776664 | 0.760512 | 0.792727 |
| 2 | 0.946750 | 0.941176 | 0.952551 | 0.779000 | 0.776471 | 0.781633 | 0.766636 | 0.760512 | 0.772727 |
| 3 | 0.946750 | 0.934804 | 0.959184 | 0.764000 | 0.752941 | 0.775510 | 0.756609 | 0.733090 | 0.780000 |
| 4 | 0.953000 | 0.941176 | 0.965306 | 0.760000 | 0.758824 | 0.761224 | 0.763902 | 0.751371 | 0.776364 |
| 5 | 0.950000 | 0.947549 | 0.952551 | 0.757000 | 0.754902 | 0.759184 | 0.777575 | 0.773309 | 0.781818 |
| Average | 0.949200 | 0.941765 | 0.956939 | 0.768400 | 0.760000 | 0.777143 | 0.768277 | 0.755759 | 0.780727 |

kernel = "poly", coef0 = 1, degree = 4

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Fold** | **train accuracy** | **train sensi** | **train speci** | **valid accuracy** | **valid sensi** | **valid speci** | **test accuracy** | **test sensi** | **test speci** |
| 1 | 0.987250 | 0.982353 | 0.992347 | 0.769000 | 0.752941 | 0.785714 | 0.764813 | 0.758684 | 0.770909 |
| 2 | 0.990250 | 0.984804 | 0.995918 | 0.764000 | 0.762745 | 0.765306 | 0.751139 | 0.744059 | 0.758182 |
| 3 | 0.989750 | 0.983333 | 0.996429 | 0.765000 | 0.747059 | 0.783673 | 0.745670 | 0.729433 | 0.761818 |
| 4 | 0.990250 | 0.985294 | 0.995408 | 0.744000 | 0.739216 | 0.748980 | 0.759344 | 0.740402 | 0.778182 |
| 5 | 0.987500 | 0.982353 | 0.992857 | 0.751000 | 0.745098 | 0.757143 | 0.757521 | 0.749543 | 0.765455 |
| Average | 0.989000 | 0.983627 | 0.994592 | 0.758600 | 0.749412 | 0.768163 | 0.755697 | 0.744424 | 0.766909 |

对比degree=2和3的情况，我们可以发现，次数的上升可以更好地拟合数据，高次多项式的拟合能力比低次多项式强是自然的。当degree都为3时，coef0增大也可以增强在训练集中的拟合能力，但略微造成了过拟合。接着我们再增大次数，我们发现当degree为4时，模型在训练集上几乎全对，但是在验证集和测试集上的表现不尽人意，说明高次多项式更容易过拟合。