监督学习

ML15

礼欣
www.python123.org

上证指数涨跌预测

数据介绍：

网易财经上获得的上证指数的历史数据，爬取了20年的上证指数数据。

实验目的：

根据给出当前时间前150天的历史数据，预测当天上证指数的涨跌。

技术路线：`sklearn.svm.SVC`

数据实例： 中核科技1997年到2017年的股票数据部分截图，红框部分为选取的特征值

日期	股票代码	名称	收盘价	最高价	最低价	开盘价	前收盘	涨跌额	涨跌幅	换手率	成交量	成交金额	总市值	流通市值
2017/1/20	'000777	中核科技	21.17	21.29	20.9	20.9	20.86	0.31	1.4861	1.0687	4097505	86664725.78	8116950444	8116950444
2017/1/19	'000777	中核科技	20.86	21.14	20.82	21.12	21.12	-0.26	-1.2311	1.0455	4008703	83926679.28	7998090990	7998090990
2017/1/18	'000777	中核科技	21.12	21.44	21.09	21.4	21.37	-0.25	-1.1699	0.922	3535002	75292556.6	8097779564	8097779564
2017/1/17	'000777	中核科技	21.37	21.49	20.75	21.17	21.15	0.22	1.0402	1.3459	5160269	109652595.5	8193633962	8193633962
2017/1/16	'000777	中核科技	21.15	22.5	20.28	22.5	22.53	-1.38	-6.1252	3.1691	12150966	261947917.1	8109282092	8109282092
2017/1/13	'000777	中核科技	22.53	22.88	22.43	22.71	22.85	-0.32	-1.4004	1.8603	7132550	161394780.8	8638398370	8638398370
2017/1/12	'000777	中核科技	22.85	23.53	22.75	23.41	23.51	-0.66	-2.8073	2.817	10800996	249876234.2	8761092000	8761092000
2017/1/11	'000777	中核科技	23.51	23.71	23.06	23.22	23.25	0.26	1.1183	4.0062	15360483	360093755.2	9014147611	9014147611
2017/1/10	'000777	中核科技	23.25	23.59	23.23	23.4	23.57	-0.32	-1.3577	2.713	10402149	243289916.6	8914459037	8914459037
2017/1/9	'000777	中核科技	23.57	23.7	22.72	22.96	23	0.57	2.4783	5.3134	20372449	475747935.6	9037152667	9037152667
2017/1/6	'000777	中核科技	23	23.19	22.82	22.95	22.87	0.13	0.5684	3.0819	11816610	271885545.4	8818604639	8818604639
2017/1/5	'000777	中核科技	22.87	22.93	22.56	22.75	22.75	0.12	0.5275	2.6699	10236812	233103957.5	8768760352	8768760352
2017/1/4	'000777	中核科技	22.75	22.81	22.54	22.65	22.6	0.15	0.6637	1.5802	6058882	137503830.2	8722750241	8722750241
2017/1/3	'000777	中核科技	22.6	22.68	22.36	22.49	22.38	0.22	0.983	1.3948	5348100	120728947.2	8665237602	8665237602
2016/12/30	'000777	中核科技	22.38	22.63	22.31	22.49	22.58	-0.2	-0.8857	1.322	5068828	113686645.3	8580885731	8580885731
2016/12/29	'000777	中核科技	22.58	22.7	22.36	22.41	22.43	0.15	0.6687	1.2307	4718858	106240524.4	8657569250	8657569250
2016/12/28	'000777	中核科技	22.43	22.72	22.42	22.63	22.58	-0.15	-0.6643	1.4301	5483427	123681991.5	8600056611	8600056611
2016/12/27	'000777	中核科技	22.58	22.93	22.56	22.92	22.91	-0.33	-1.4404	1.5646	5998804	136263536.3	8657569250	8657569250
2016/12/26	'000777	中核科技	22.91	22.96	22.38	22.7	22.89	0.02	0.0874	2.1045	8068925	182955263.2	8784097056	8784097056
2016/12/23	'000777	中核科技	22.89	23.25	22.64	22.95	23.11	-0.22	-0.952	2.38	9125180	208889546.6	8776428704	8776428704
2016/12/22	'000777	中核科技	23.11	23.55	22.75	22.82	22.82	0.29	1.2708	3.7389	14335433	333074476.8	8860780574	8860780574
2016/12/21	'000777	中核科技	22.82	22.96	22.58	22.59	22.53	0.29	1.2872	2.2115	8479447	193133942.4	8749589472	8749589472
2016/12/20	'000777	中核科技	22.53	22.67	22.41	22.67	22.69	-0.16	-0.7052	1.329	5095772	114711037.5	8638398370	8638398370
2016/12/19	'000777	中核科技	22.69	22.77	22.51	22.67	22.63	0.06	0.2651	1.4709	5639790	127588225.1	8699745185	8699745185
2016/12/16	'000777	中核科技	22.63	22.88	22.58	22.73	22.71	-0.08	-0.3523	1.9302	7400685	168016411.1	8676740130	8676740130

实验过程：

- 使用算法： SVM，
- 实现过程：

1. 建立工程，导入sklearn相关包

```
import pandas as pd
import numpy as np
from sklearn import svm
from sklearn import cross_validation
```

关于一些相关包的介绍：

- pandas：用来加载CSV数据的工具包
- numpy：支持高级大量的维度数组与矩阵运算，此外也针对数组运算提供大量的数学函数库。
- sklearn下svm：SVM算法
- sklearn下cross_validation：交叉验证

2. 数据加载&&数据预处理

```
data=pd.read_csv('stock/000777.csv',encoding='gbk',parse_dates=[0],index_col=0)
data.sort_index(0,ascending=True,inplace=True)

dayfeature=150
featurenum=5*dayfeature
x=np.zeros((data.shape[0]-dayfeature,featurenum+1))
y=np.zeros((data.shape[0]-dayfeature))
```

一些参数解释：读入数据

pd：pandas包的实例参数

read_csv()：详细解释（http://pandas.pydata.org/pandas-docs/stable/generated/pandas.read_csv.html）

pandas.read_csv（数据源，encoding=编码格式为gbk， parse_dates=第0列解析为日期， index_col=用作行索引的列编号）

sort_index()：详细解释（ http://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.sort_index.html ）

DataFrame.sort_index(axis=0 (按0列排), ascending=True（升序），inplace=False（排序后是否覆盖原数据））data 按照时间升序排列

2. 数据加载&&数据预处理

```
data=pd.read_csv('stock/000777.csv',encoding='gbk',parse_dates=[0],index_col=0)
data.sort_index(0,ascending=True,inplace=True)

dayfeature=150
featurenum=5*dayfeature
x=np.zeros((data.shape[0]-dayfeature,featurenum+1))
y=np.zeros((data.shape[0]-dayfeature))
```

参数解释：

选取5列数据作为特征：收盘价 最高价 最低价 开盘价 成交量

dayfeature：选取150天的数据

featurenum：选取的5个特征*天数

x：记录150天的5个特征值

y：记录涨或者跌

data.shape[0]-dayfeature意思是因为我们要用150天数据做训练，对于条目为200条的数据，只有50条数据是有前150天的数据来训练的，所以训练集的大小就是200-150， 对于每一条数据，他的特征是前150天的所有特征数据，即150*5， +1是将当天的开盘价引入作为一条特征数据

2. 数据加载&&数据预处理

	特征1	特征2	特征3	特征4	特征5
第1天					
第2天					
第3天					
……					

150天

150天

X示意：

150天

{v1,v2,v3,v4,v5},
{v1,v2,v3,v4,v5}……{v1,v2,v3,v4,v5}

……

150天

{v1,v2,v3,v4,v5},
{v1,v2,v3,v4,v5}……{v1,v2,v3,v4,v5}

data.shape[0]-dayfeature

2. 数据加载&&数据预处理

```
for i in range(0,data.shape[0]-dayfeature):
    x[i,0:featurenum]=np.array(data[i:i+dayfeature]\
            [[u'收盘价',u'最高价',u'最低价',u'开盘价',u'成交量']]).reshape((1,featurenum))
            //将数据中的“收盘价” “最高价” “开盘价” “成交量”存入x数组中
    x[i,featurenum]=data.ix[i+dayfeature][u'开盘价']
            //最后一列记录当日的开盘价
```

```
for i in range(0,data.shape[0]-dayfeature):
    if data.ix[i+dayfeature][u'收盘价']>=data.ix[i+dayfeature][u'开盘价']:
        y[i]=1
    else:
        y[i]=0
```

//如果当天收盘价高于开盘价，y[i]=1代表涨，0代表跌

参数解释：

u:unicode编码

reshape:转换成1行，featurenum列

ix :索引

3. 创建SVM并进行交叉验证

```
clf=svm.SVC(kernel='rbf')
//调用svm函数，并设置kernel参数，默认是rbf，其它： ‘linear’ ‘poly’ ‘sigmoid’
result = []
for i in range(5):
    x_train, x_test, y_train, y_test = cross_validation.train_test_split(x, y, test_size = 0.2)
//x和y的验证集和测试集，切分80-20%的测试集
    clf.fit(x_train, y_train)
//训练数据进行训练
    result.append(np.mean(y_test == clf.predict(x_test)))
//将预测数据和测试集的验证数据比对
print("svm classifier accuacy:")

print(result)
```

实验结果：

核函数	1	2	3	4	5
rbf	0.5320	0.5287	0.5504	0.5374	0.5352
sigmoid	0.5418	0.5472	0.5363	0.5418	0.5537

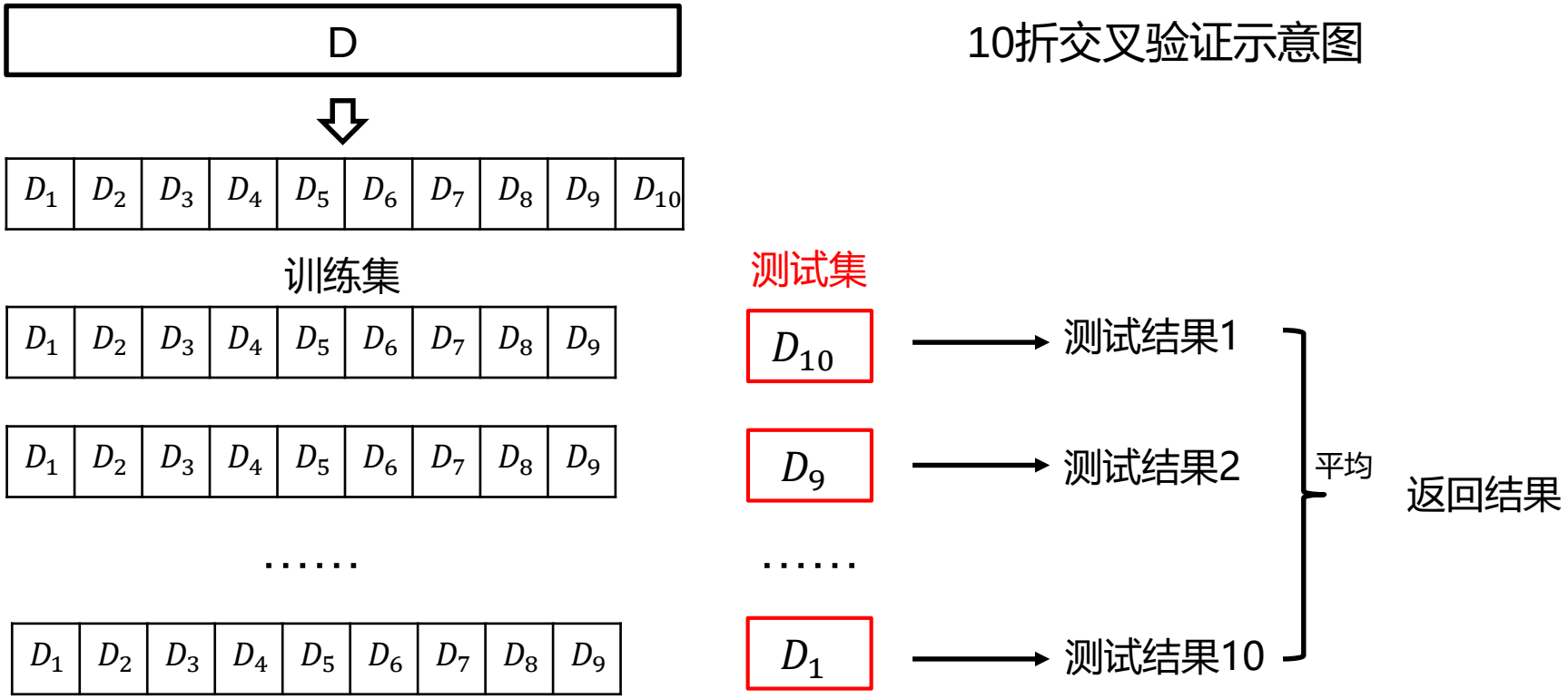
本次实验运用了两个核函数做实验，准确率由表中数据所示。5次交叉验证的准确率相近，均为53%左右。

交叉验证

基本思想：

交叉验证法先将数据集D划分为k个大小相似的互斥子集，每个自己都尽可能保持数据分布的一致性，即从D中通过分层采样得到。然后，每次用k-1个子集的并集作为训练集，余下的那个子集作为测试集；这样就可获得k组训练/测试集，从而可进行k次训练和测试，最终返回的是这个k个测试结果的均值。通常把交叉验证法称为“k者交叉验证”，k最常用的取值是10，此时称为10折交叉验证。

交叉验证



10折交叉验证示意图