poker 例
每个实例是一手牌（5 张），每张牌用两个特征（花色 1-4，点数 1-13）描述，所以共有 10 个特征。需要将特征进行 hot 编码，转换成 0/1 二元特征。
实例对应的类别共有 10 类，分别用 0-9 表示。例如 0 表示一手乱牌，1 表示有一对点数相同的牌，2 表示有两对，。。。，8 表示同花顺，9 表示同花大顺（10-J-Q-K-A）。

```
import urllib2

from sklearn.cross_validation import cross_val_score

from sklearn.svm import SVC

import numpy as np

from sklearn.datasets import load_svmlight_file

url =
'http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/multiclas
s/poker.bz2'

with open('d:/datasci/poker.bz2','wb') as W:
    W.write(urllib2.urlopen(url).read())

X,y = load_svmlight_file('d:/datasci/poker.bz2')

X.shape
(25010, 10)

type(X)
scipy.sparse.csr.csr_matrix

type(y)
numpy.ndarray

X1 = X.toarray()

type(X1)
numpy.ndarray

X1.shape
(25010, 10)

X1[:4]
array([[ 1., 10.,  1., 11.,  1., 13.,  1., 12.,  1.,  1.]])
```

```
       [ 2., 11.,  2., 13.,  2., 10.,  2., 12.,  2.,  1.],
       [ 3., 12.,  3., 11.,  3., 13.,  3., 10.,  3.,  1.],
       [ 4., 10.,  4., 11.,  4.,  1.,  4., 13.,  4., 12.]])

y[:4]
array([9., 9., 9., 9.])

from sklearn.preprocessing import OneHotEncoder

hot_enc = OneHotEncoder(sparse=True)

# 对 X1 进行 OneHot 编码（相当于虚拟变量）

X_train = hot_enc.fit_transform(X1)

type(X_train)
scipy.sparse.csr.csr_matrix

X2 = X_train.toarray()

X2.shape
(25010, 85)

X2[:1]
[1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0.,
 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 0.,
 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1.,
 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 0.,
 1., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.]

from sklearn.svm import LinearSVC

lsvc = LinearSVC(dual=False)

lsvc.fit(X_train,y)

y_predicted = lsvc.predict(X_train)

pd.Series(y_predicted).value_counts()
0.0    23338
1.0     1672

pd.Series(y).value_counts()
0.0    12493
```

```
1.0   10599
2.0    1206
3.0     513
4.0      93
5.0      54
6.0      36
7.0       6
9.0       5
8.0       5

# 看看被预测为 1 的样本的实际类别
res = []
for i in np.arange(25010):
    if y_predicted[i]==1:
        res.append(y[i])

pd.Series(res).value_counts()
1.0   796
0.0   751
2.0    79
3.0    33
4.0     5
5.0     3
7.0     2
9.0     1
8.0     1
6.0     1
# 模型应该是按 0 vs {1-9}分类的

y[:4]
array([9., 9., 9., 9.])

y_predicted[:4]
array([0., 1., 0., 0.])

lsvc.score(X_train,y)
0.5013194722111155

scores = cross_val_score(lsvc,X_train,y,
                         cv=3,scoring='accuracy',n_jobs=-1)

print "mean = %0.3f, std = %0.3f" %
       (np.mean(scores),np.std(scores))
mean = 0.490, std = 0.004
```