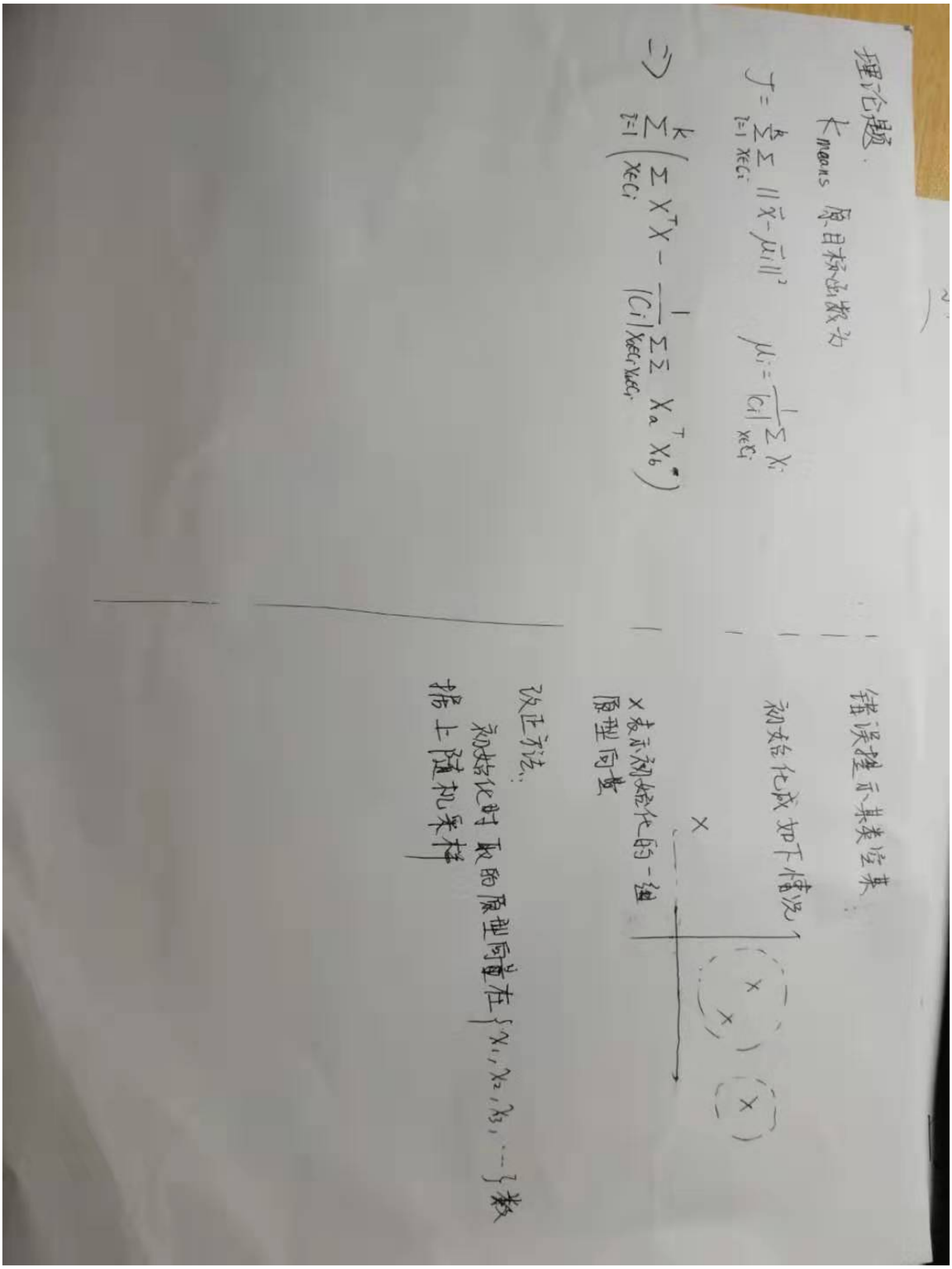
理论题



理论题.

K_{means} 原目标函数为

$$J=\sum_{i=1}^{k}\sum_{x\in C_i}\|\vec{x}-\vec{\mu_i}\|^2 \qquad \mu_i=\frac{1}{|C_i|}\sum_{x\in C_i}x_i$$

$$\Rightarrow \sum_{i=1}^{k}\left(\sum_{x\in C_i}X^TX-\frac{1}{|C_i|}\sum_{x_a\in C_i}\sum_{x_b\in C_i}X_a^TX_b\right)$$

错误提示某类空集

初始化成如下情况

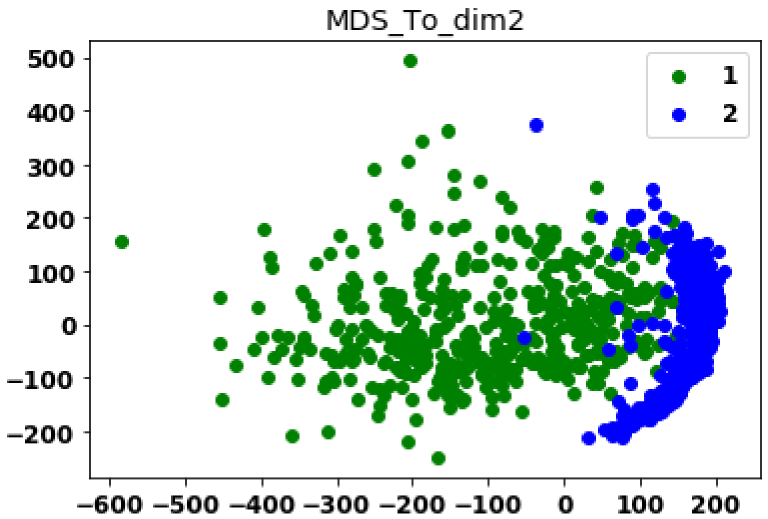
x表示初始化的一组原型向量

改正方法:

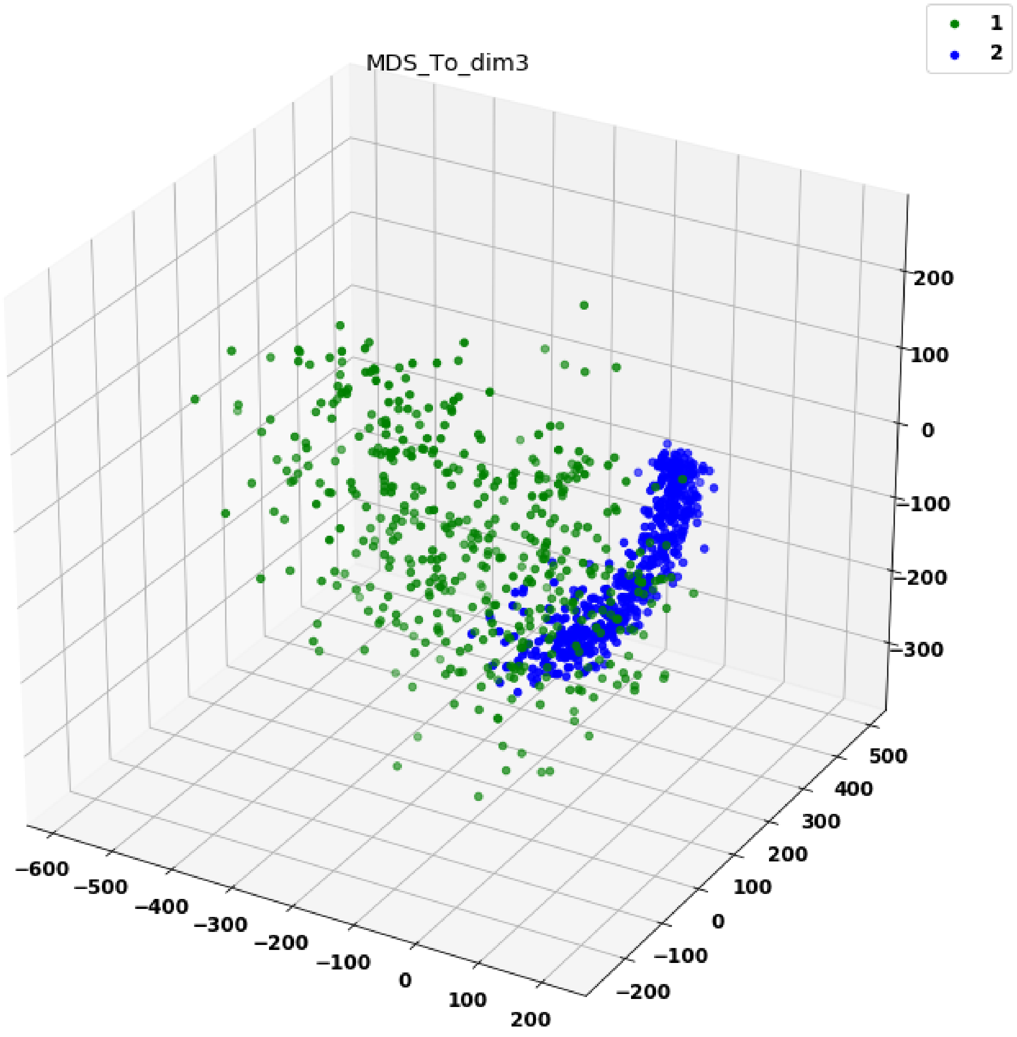
初始化时取的原型向量在$\{x_1,x_2,x_3,\cdots\}$数据上随机采样

编程结果

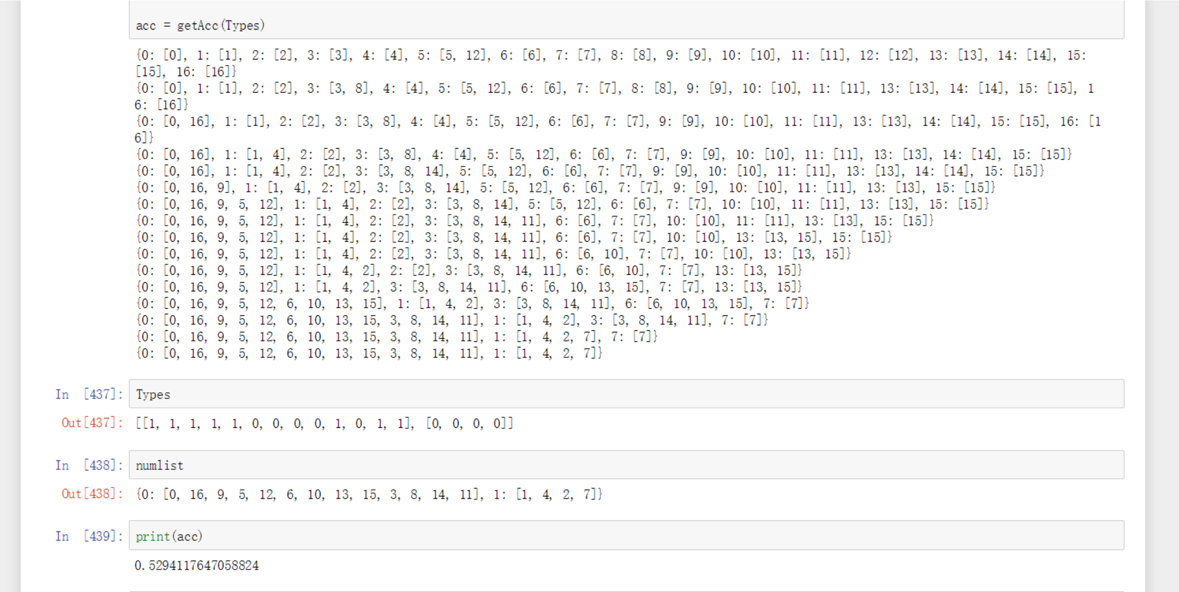
MDS降维2维



MDS降维3维



欧氏距离



```
acc = getAcc(Types)
```

```
{0: [0], 1: [1], 2: [2], 3: [3], 4: [4], 5: [5, 12], 6: [6], 7: [7], 8: [8], 9: [9], 10: [10], 11: [11], 12: [12], 13: [13], 14: [14], 15:
[15], 16: [16]}
{0: [0], 1: [1], 2: [2], 3: [3, 8], 4: [4], 5: [5, 12], 6: [6], 7: [7], 8: [8], 9: [9], 10: [10], 11: [11], 13: [13], 14: [14], 15: [15], 1
6: [16]}
{0: [0, 16], 1: [1], 2: [2], 3: [3, 8], 4: [4], 5: [5, 12], 6: [6], 7: [7], 9: [9], 10: [10], 11: [11], 13: [13], 14: [14], 15: [15], 16: [1
6]}
{0: [0, 16], 1: [1, 4], 2: [2], 3: [3, 8], 4: [4], 5: [5, 12], 6: [6], 7: [7], 9: [9], 10: [10], 11: [11], 13: [13], 14: [14], 15: [15]}
{0: [0, 16], 1: [1, 4], 2: [2], 3: [3, 8, 14], 5: [5, 12], 6: [6], 7: [7], 9: [9], 10: [10], 11: [11], 13: [13], 14: [14], 15: [15]}
{0: [0, 16, 9], 1: [1, 4], 2: [2], 3: [3, 8, 14], 5: [5, 12], 6: [6], 7: [7], 9: [9], 10: [10], 11: [11], 13: [13], 15: [15]}
{0: [0, 16, 9, 5, 12], 1: [1, 4], 2: [2], 3: [3, 8, 14], 5: [5, 12], 6: [6], 7: [7], 10: [10], 11: [11], 13: [13], 15: [15]}
{0: [0, 16, 9, 5, 12], 1: [1, 4], 2: [2], 3: [3, 8, 14, 11], 6: [6], 7: [7], 10: [10], 11: [11], 13: [13], 15: [15]}
{0: [0, 16, 9, 5, 12], 1: [1, 4], 2: [2], 3: [3, 8, 14, 11], 6: [6], 7: [7], 10: [10], 13: [13, 15], 15: [15]}
{0: [0, 16, 9, 5, 12], 1: [1, 4], 2: [2], 3: [3, 8, 14, 11], 6: [6, 10], 7: [7], 10: [10], 13: [13, 15]}
{0: [0, 16, 9, 5, 12], 1: [1, 4, 2], 2: [2], 3: [3, 8, 14, 11], 6: [6, 10], 7: [7], 13: [13, 15]}
{0: [0, 16, 9, 5, 12], 1: [1, 4, 2], 3: [3, 8, 14, 11], 6: [6, 10, 13, 15], 7: [7], 13: [13, 15]}
{0: [0, 16, 9, 5, 12, 6, 10, 13, 15], 1: [1, 4, 2], 3: [3, 8, 14, 11], 6: [6, 10, 13, 15], 7: [7]}
{0: [0, 16, 9, 5, 12, 6, 10, 13, 15, 3, 8, 14, 11], 1: [1, 4, 2], 3: [3, 8, 14, 11], 7: [7]}
{0: [0, 16, 9, 5, 12, 6, 10, 13, 15, 3, 8, 14, 11], 1: [1, 4, 2, 7], 7: [7]}
{0: [0, 16, 9, 5, 12, 6, 10, 13, 15, 3, 8, 14, 11], 1: [1, 4, 2, 7]}
```

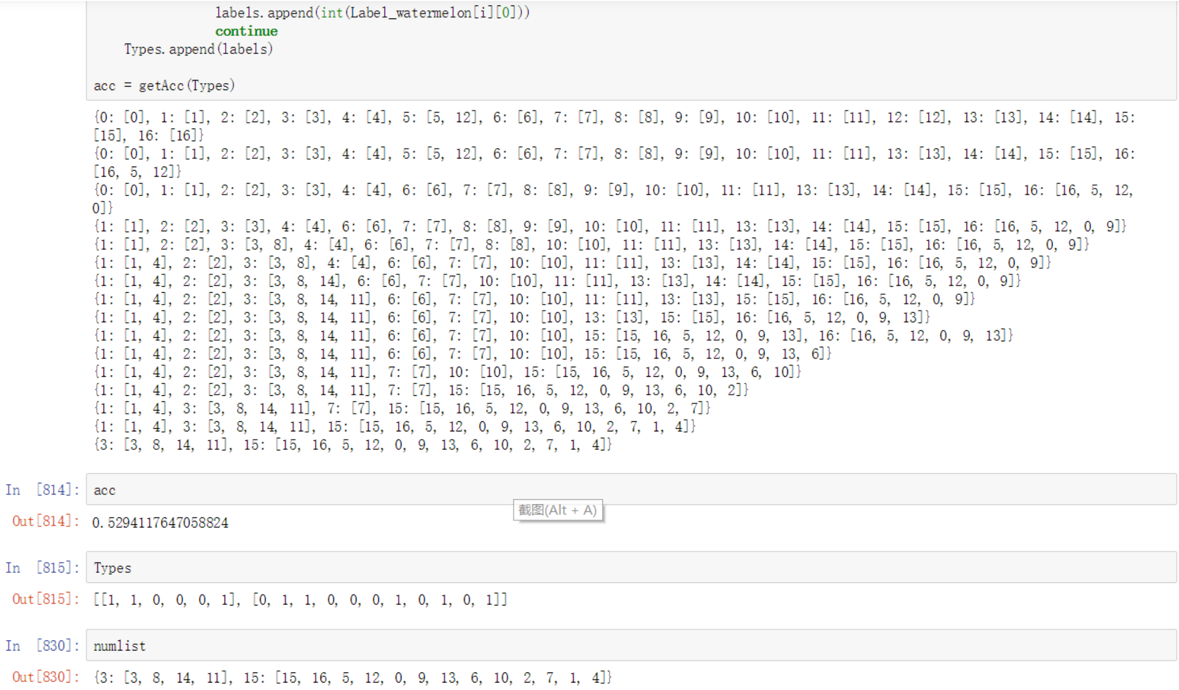
```
In  [437]: Types
Out[437]: [[1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 0, 1, 1], [0, 0, 0, 0]]

In  [438]: numlist
Out[438]: {0: [0, 16, 9, 5, 12, 6, 10, 13, 15, 3, 8, 14, 11], 1: [1, 4, 2, 7]}

In  [439]: print(acc)

           0.5294117647058824
```

Ncut距离



```
                labels.append(int(Label_watermelon[i][0]))
                continue
        Types.append(labels)

acc = getAcc(Types)
```

```
{0: [0], 1: [1], 2: [2], 3: [3], 4: [4], 5: [5, 12], 6: [6], 7: [7], 8: [8], 9: [9], 10: [10], 11: [11], 12: [12], 13: [13], 14: [14], 15:
[15], 16: [16]}
{0: [0], 1: [1], 2: [2], 3: [3], 4: [4], 5: [5, 12], 6: [6], 7: [7], 8: [8], 9: [9], 10: [10], 11: [11], 13: [13], 14: [14], 15: [15], 16:
[16, 5, 12]}
{0: [0], 1: [1], 2: [2], 3: [3], 4: [4], 6: [6], 7: [7], 8: [8], 9: [9], 10: [10], 11: [11], 13: [13], 14: [14], 15: [15], 16: [16, 5, 12,
0]}
{1: [1], 2: [2], 3: [3], 4: [4], 6: [6], 7: [7], 8: [8], 9: [9], 10: [10], 11: [11], 13: [13], 14: [14], 15: [15], 16: [16, 5, 12, 0, 9]}
{1: [1], 2: [2], 3: [3, 8], 4: [4], 6: [6], 7: [7], 8: [8], 10: [10], 11: [11], 13: [13], 14: [14], 15: [15], 16: [16, 5, 12, 0, 9]}
{1: [1, 4], 2: [2], 3: [3, 8], 4: [4], 6: [6], 7: [7], 10: [10], 11: [11], 13: [13], 14: [14], 15: [15], 16: [16, 5, 12, 0, 9]}
{1: [1, 4], 2: [2], 3: [3, 8, 14], 6: [6], 7: [7], 10: [10], 11: [11], 13: [13], 14: [14], 15: [15], 16: [16, 5, 12, 0, 9]}
{1: [1, 4], 2: [2], 3: [3, 8, 14, 11], 6: [6], 7: [7], 10: [10], 11: [11], 13: [13], 15: [15], 16: [16, 5, 12, 0, 9]}
{1: [1, 4], 2: [2], 3: [3, 8, 14, 11], 6: [6], 7: [7], 10: [10], 13: [13], 15: [15], 16: [16, 5, 12, 0, 9, 13]}
{1: [1, 4], 2: [2], 3: [3, 8, 14, 11], 6: [6], 7: [7], 10: [10], 15: [15, 16, 5, 12, 0, 9, 13], 16: [16, 5, 12, 0, 9, 13]}
{1: [1, 4], 2: [2], 3: [3, 8, 14, 11], 6: [6], 7: [7], 10: [10], 15: [15, 16, 5, 12, 0, 9, 13, 6]}
{1: [1, 4], 2: [2], 3: [3, 8, 14, 11], 7: [7], 10: [10], 15: [15, 16, 5, 12, 0, 9, 13, 6, 10]}
{1: [1, 4], 2: [2], 3: [3, 8, 14, 11], 7: [7], 15: [15, 16, 5, 12, 0, 9, 13, 6, 10, 2]}
{1: [1, 4], 3: [3, 8, 14, 11], 7: [7], 15: [15, 16, 5, 12, 0, 9, 13, 6, 10, 2, 7]}
{1: [1, 4], 3: [3, 8, 14, 11], 15: [15, 16, 5, 12, 0, 9, 13, 6, 10, 2, 7, 1, 4]}
{3: [3, 8, 14, 11], 15: [15, 16, 5, 12, 0, 9, 13, 6, 10, 2, 7, 1, 4]}
```

```
In  [814]: acc
Out[814]: 0.5294117647058824

In  [815]: Types
Out[815]: [[1, 1, 0, 0, 0, 1], [0, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1]]

In  [830]: numlist
Out[830]: {3: [3, 8, 14, 11], 15: [15, 16, 5, 12, 0, 9, 13, 6, 10, 2, 7, 1, 4]}
```

树

```
In  [793]: np.random.seed(11)
           watermalon = pd.read_csv("watermalon.csv",engine="python")
           Treetrainset, Treetestset = getWaterMelonDe(watermalon,rate=0.7)
```

```
In  [794]: tree = InformationTree()
```

```
In  [795]: tree.train(Treetrainset)
```

```
脐部
根蒂
稍蜷 Bad
蜷缩 Good
平坦 Bad
敲声
沉闷 Bad
浊响 Good
```

```
In  [796]: tree.validate(Treetestset)
```

```
0.8333333333333334
```

Out[796]:

	色泽	根蒂	敲声	纹理	脐部	触感	好瓜	预测
11	乌黑	稍蜷	浊响	清晰	稍凹	软粘	否	是
12	青绿	稍蜷	浊响	稍糊	凹陷	硬滑	否	否
13	乌黑	稍蜷	浊响	清晰	稍凹	硬滑	是	是
14	乌黑	蜷缩	沉闷	清晰	凹陷	硬滑	是	是
15	浅白	蜷缩	浊响	模糊	平坦	软粘	否	否
16	青绿	蜷缩	沉闷	稍糊	稍凹	硬滑	否	否