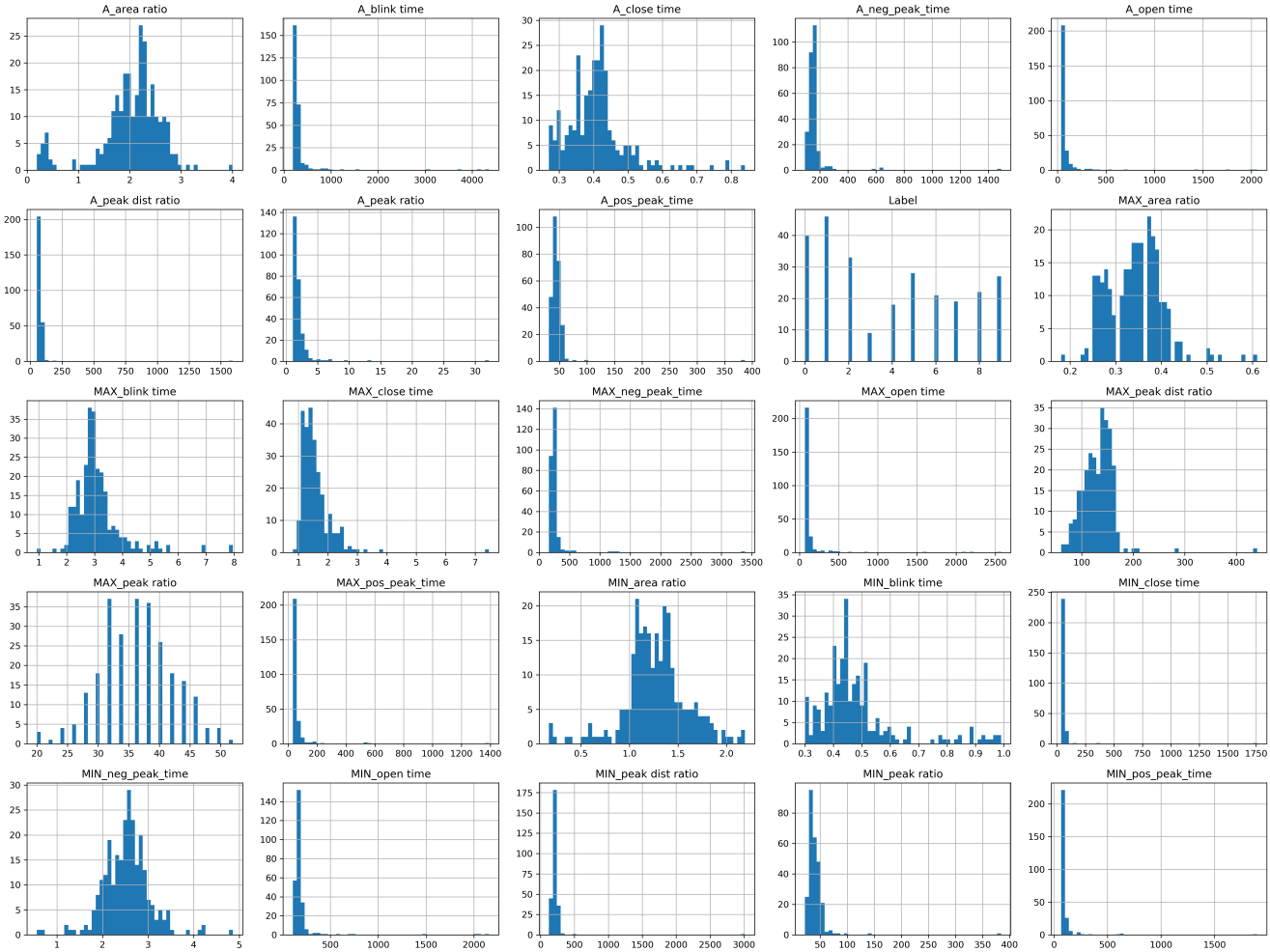
柱状图分析


A_area ratio
A_blink time
A_close time
A_neg_peak_time
A_open time
A_peak dist ratio
A_peak ratio
A_pos_peak_time
Label
MAX_area ratio
MAX_blink time
MAX_close time
MAX_neg_peak_time
MAX_open time
MAX_peak dist ratio
MAX_peak ratio
MAX_pos_peak_time
MIN_area ratio
MIN_blink time
MIN_close time
MIN_neg_peak_time
MIN_open time
MIN_peak dist ratio
MIN_peak ratio
MIN_pos_peak_time

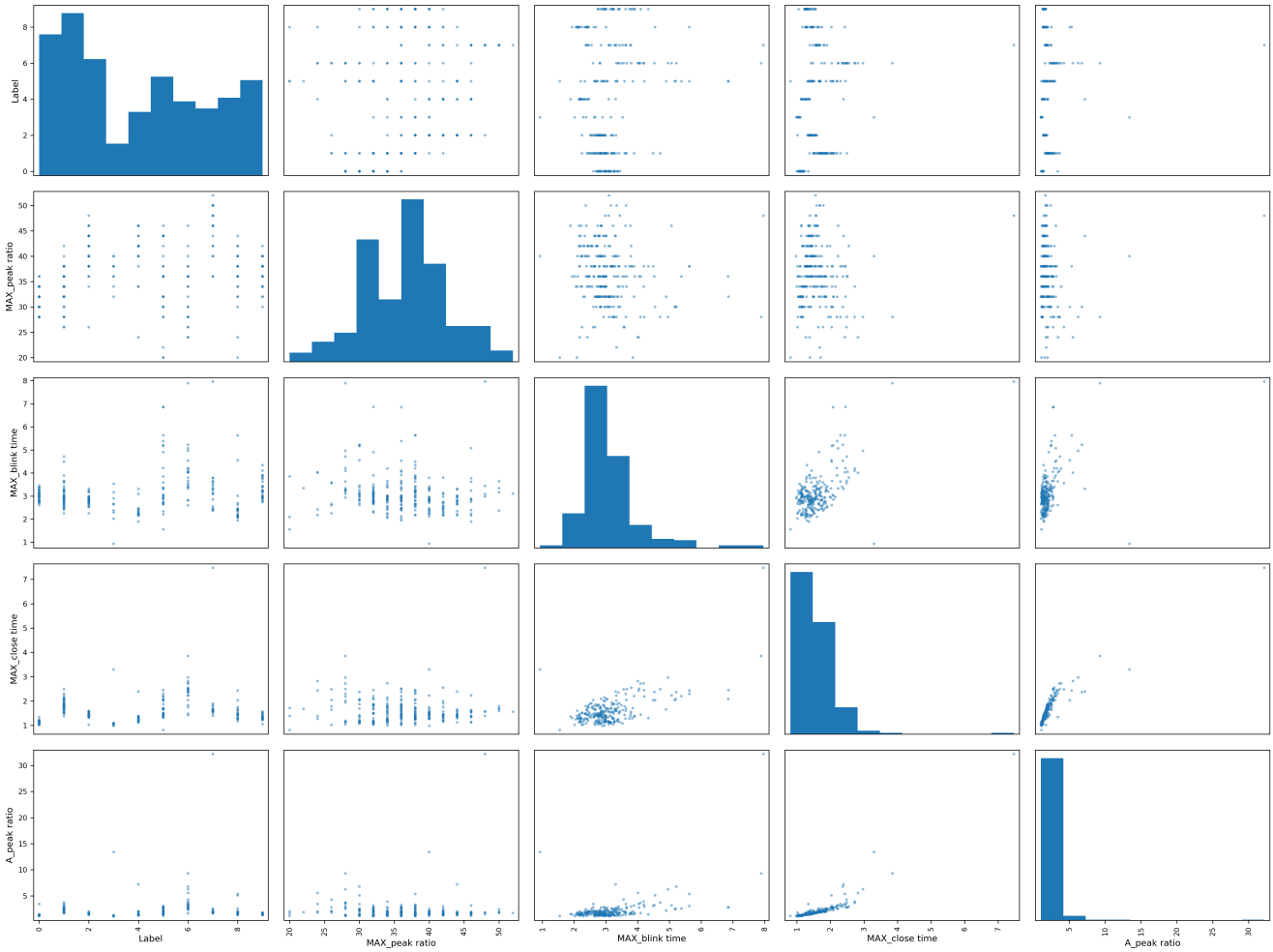
相关性分析

In [11]:

```
corr_matrix = data.corr()
corr_matrix["Label"].sort_values(ascending=False)
```

Out[11]:

```
Label                  1.000000
MAX_peak ratio         0.242459
MAX_blink time         0.144644
MAX_close time         0.142896
A_peak ratio           0.111361
MIN_neg_peak_time      0.109658
A_pos_peak_time        0.093964
MIN_area ratio         0.064174
A_area ratio           0.052850
MAX_peak dist ratio    0.028506
A_peak dist ratio      0.018295
MIN_close time         0.017658
MIN_peak dist ratio    0.011860
MIN_blink time         0.010979
A_neg_peak_time        0.005635
MIN_peak ratio        -0.000321
MAX_neg_peak_time     -0.002566
MIN_open time         -0.007667
MIN_pos_peak_time     -0.008977
A_blink time          -0.021028
A_close time          -0.022535
MAX_pos_peak_time     -0.024888
A_open time           -0.026971
MAX_open time         -0.032130
MAX_area ratio        -0.060555
Name: Label, dtype: float64
```



测试集与训练集制作

根据Label的比例进行分层抽取 在测试集中设置 k-fold-cv 这里用 7

confusion_matrix

训练集

```
array([[25,  0,  2,  3,  0,  1,  0,  0,  1,  0],
       [ 1, 28,  0,  0,  0,  5,  1,  0,  2,  0],
       [ 2,  0, 22,  0,  2,  0,  0,  0,  0,  0],
       [ 0,  0,  1,  3,  0,  0,  1,  0,  0,  2],
       [ 0,  4,  0,  0,  7,  2,  0,  0,  0,  1],
       [ 0,  2,  0,  1,  2,  9,  2,  5,  1,  0],
       [ 0,  1,  0,  2,  0,  0, 12,  1,  1,  0],
       [ 1,  1,  0,  2,  1,  0,  2,  6,  1,  1],
       [ 1,  3,  0,  0,  0,  1,  1,  0, 12,  0],
       [ 1,  0,  1,  5,  0,  0,  0,  0,  1, 14]], dtype=int64)
```

测试集

```
array([[7, 0, 0, 0, 0, 0, 0, 0, 0, 1],
       [0, 9, 0, 0, 0, 0, 0, 0, 0, 0],
       [0, 0, 7, 0, 0, 0, 0, 0, 0, 0],
       [1, 0, 0, 1, 0, 0, 0, 0, 0, 0],
       [0, 0, 0, 0, 3, 0, 1, 0, 0, 0],
       [0, 1, 1, 1, 0, 3, 0, 0, 0, 0],
       [0, 1, 1, 0, 0, 0, 2, 0, 0, 0],
       [0, 1, 0, 0, 0, 0, 0, 3, 0, 0],
       [0, 0, 0, 0, 0, 0, 0, 1, 3, 0],
       [1, 0, 1, 0, 0, 0, 0, 0, 0, 3]], dtype=int64)
```

precision 与 recall

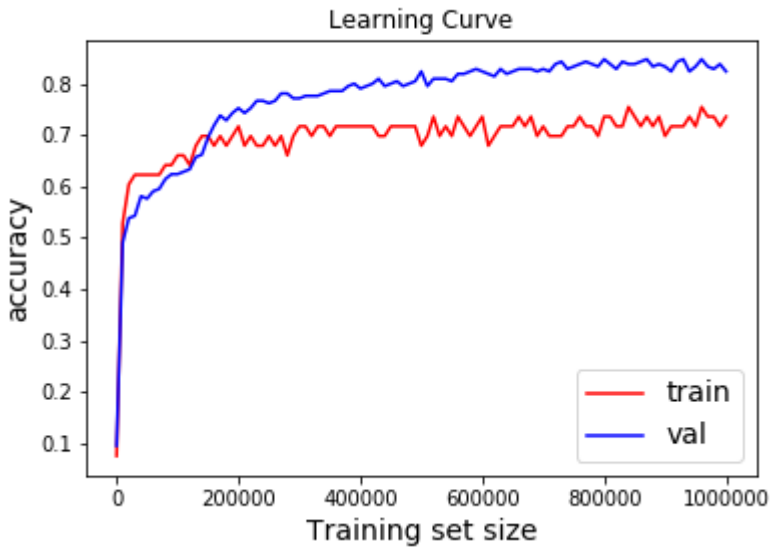
训练集

	precision	recall	f1-score	support
0.0	0.81	0.78	0.79	32
1.0	0.72	0.76	0.74	37
2.0	0.85	0.85	0.85	26
3.0	0.19	0.43	0.26	7
4.0	0.58	0.50	0.54	14
5.0	0.50	0.41	0.45	22
6.0	0.63	0.71	0.67	17
7.0	0.50	0.40	0.44	15
8.0	0.63	0.67	0.65	18
9.0	0.78	0.64	0.70	22
micro avg	0.66	0.66	0.66	210
macro avg	0.62	0.61	0.61	210
weighted avg	0.67	0.66	0.66	210

测试集

	precision	recall	f1-score	support
0.0	0.78	0.88	0.82	8
1.0	0.75	1.00	0.86	9
2.0	0.70	1.00	0.82	7
3.0	0.50	0.50	0.50	2
4.0	1.00	0.75	0.86	4
5.0	1.00	0.50	0.67	6
6.0	0.67	0.50	0.57	4
7.0	0.75	0.75	0.75	4
8.0	1.00	0.75	0.86	4
9.0	0.75	0.60	0.67	5
micro avg	0.77	0.77	0.77	53
macro avg	0.79	0.72	0.74	53
weighted avg	0.80	0.77	0.76	53

MLP Learning Curve



在加入 l2 正则化之后，过拟合情况有所缓解。