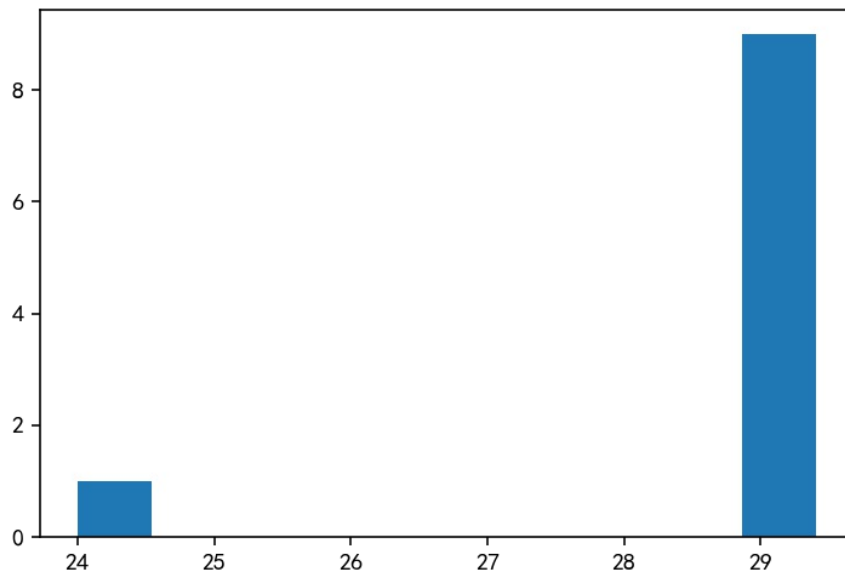# 小数据
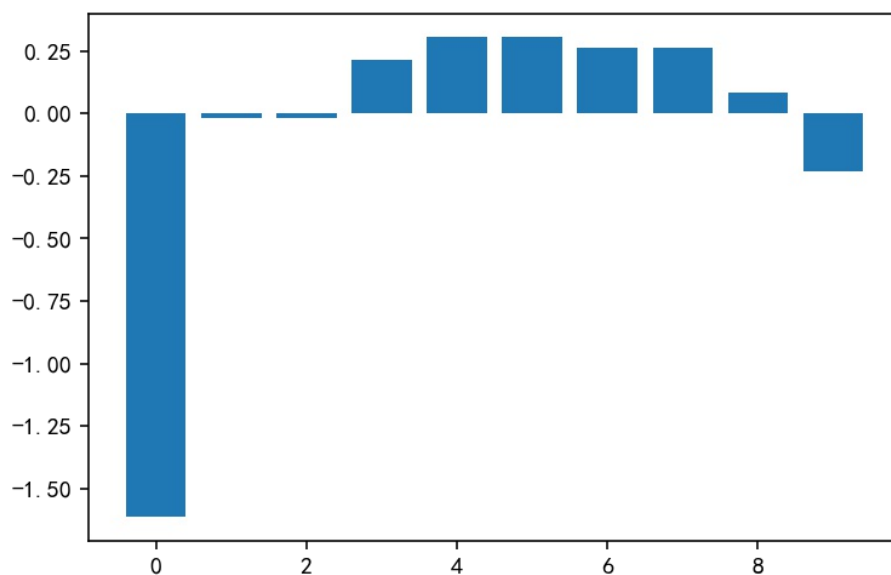
```
data = np.array([24.0, 28.9, 28.9, 29.0, 29.1, 29.1, 29.2, 29.2, 29.3, 29
```

```
n,bins,_ = plt.hist(data, bins=10)
```



```
from sklearn.neighbors.kde import KernelDensity

kde = KernelDensity(kernel='gaussian', bandwidth=0.2).fit(data.reshape((-
score = kde.score_samples(data.reshape((-1,1)))
plt.bar(range(len(score)), score);
```
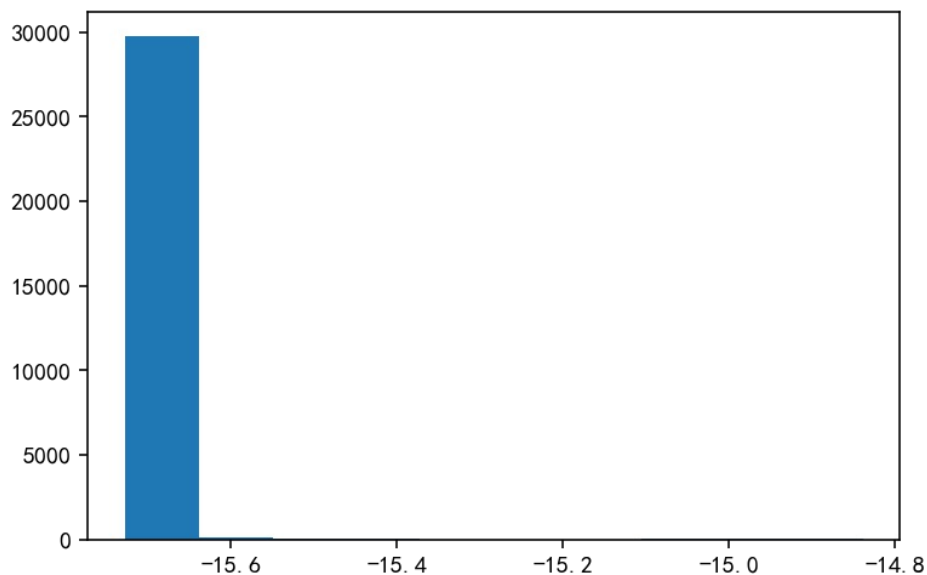

信用卡数据

```python
data = pd.read_excel('default of credit card clients.xls', header=1,index
```

```python
kde = KernelDensity(kernel='gaussian', bandwidth=0.5).fit(data)
score = kde.score_samples(data)
```

```python
plt.hist(score);
```


对于小于**99%**的判定为异常

```python
out_line = np.percentile(score,99)
```

```python
y_pred = np.zeros(len(data))
y_pred[score > out_line] = 1
```

```python
from sklearn.metrics import classification_report
print(classification_report(data[data.columns[-1]].values, y_pred))
```

```
             precision    recall  f1-score   support

          0       0.78      0.99      0.87     23364
          1       0.31      0.01      0.03      6636

avg / total       0.68      0.78      0.69     30000
```