

1 垂直平分分类器

In [1]:

```
import numpy as np
import matplotlib.pyplot as plt
import matplotlib as mpl
mpl.rcParams['font.sans-serif'] = ['SimHei'] # 中文
plt.rcParams['axes.unicode_minus'] = False # 正负号
```

executed in 396ms, finished 16:19:24 2021-03-12

1.1 生成实验数据

In [2]:

```
mean = np.array([3, 3])
cov = np.array([[3.5, -0.5], [-0.5, 3.5]])
```

executed in 9ms, finished 16:19:24 2021-03-12

In [3]:

```
cluster_1 = np.random.multivariate_normal(mean, cov, (200,), 'raise')
cluster_2 = np.random.multivariate_normal(-mean, cov, (200,), 'raise')
```

executed in 138ms, finished 16:19:24 2021-03-12

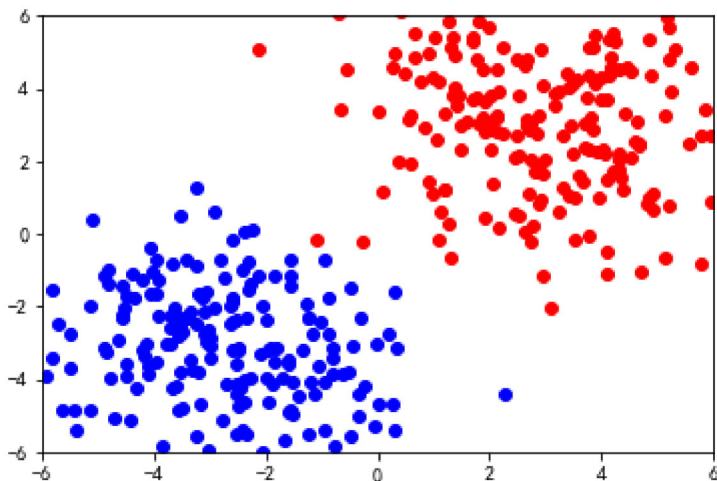
In [4]:

```
plt.scatter(cluster_1[:, 0], cluster_1[:, 1], color='r')
plt.scatter(cluster_2[:, 0], cluster_2[:, 1], color='b')
plt.xlim(-6, 6)
plt.ylim(-6, 6)
```

executed in 290ms, finished 16:19:24 2021-03-12

Out[4]:

(-6, 6)



1.2 生成测试集

In [5]:

```
test = np.random.uniform([-6, -6], [6, 6], [200, 2])
```

executed in 6ms, finished 16:19:24 2021-03-12

1.3 实验结果

In [6]:

```
center_1 = np.mean(cluster_1, axis=0)  
center_2 = np.mean(cluster_2, axis=0)
```

executed in 153ms, finished 16:19:25 2021-03-12

In [7]:

```
plt.scatter(center_1[0], center_1[1], color='black')  
plt.text (center_1[0], center_1[1], '第一类中心')  
plt.scatter(center_2[0], center_2[1], color='black')  
plt.text (center_2[0], center_2[1], '第二类中心')  
▼ for t in test:  
    d1 = np.linalg.norm(t-center_1)  
    d2 = np.linalg.norm(t-center_2)  
▼    if d1 < d2:  
        plt.scatter(t[0], t[1], color='r')  
▼    else:  
        plt.scatter(t[0], t[1], color='b')
```

executed in 991ms, finished 16:19:26 2021-03-12

