In [13]:

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from scipy.spatial import distance
from sklearn.metrics import accuracy_score
from sklearn.neighbors import KNeighborsClassifier
import matplotlib.pyplot as plt

data = pd.read_csv('iris.csv')

x = data.values[:, :4]
y = np.zeros(150)

for i in range(len(y)):
    if data.values[i, 4]=='setosa':
        y[i] = 0
    elif data.values[i, 4]=='versicolor':
        y[i] = 1
    elif data.values[i, 4]=='virginica':
        y[i] = 2

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.33,
random_state=42)
d = distance.cdist(x_test,x_train,'euclidean')


for i in range (50):
    y1=np.argsort(d,axis=1)
j=int(7)
k=int(0)
y_predict=[]
for j in range(0,50):
    y2 = np.zeros(3)
    for i in range (0,j):
        inside=int(y1[k][i])
        value=int(y_train[inside])
        y2[value]+=1

    y_predict.append(np.argmax(y2, axis=0))
accuracy_score(y_test, y_predict)
```

Out[13]:

0.28

In [14]:

```
k2 = np.array([1,2,5,7,9,11,13,15])
result=[]
r=int(0)
for k1 in k2:
    y_predict=[]
    for j in range(0,50):
        y2 = np.zeros(3)
```

```
        for i in range (0,k1):
            inde=int(y1[j][i])
            value=int(y_train[inde])
            y2[value]+=1

        y_predict.append(np.argmax(y2, axis=0))
    result.append(accuracy_score(y_test,y_predict))
final=np.array(result)
plt.plot(final,k2)
plt.show()
```