

Report

k-NN 분류를 이용한 당도에 의한 과일의 분류



학과	수학과
학번	2018010705
이름	신영민

<목차>

I . 문제 정의

II . 데이터 준비

III. 구현 및 실험

IV. 결과 분석

V. 결론 도출

VI. 출처

I. 문제정의

1. 동기

우리 주변에는 다양한 과일들이 있습니다. 과일들을 생각해보면 ‘달다’라는 생각이 듭니다. 과일들은 달지만 다 같은 맛을 내는 것은 아닙니다. 그렇다면 맛을 달게 느끼게 하는 물질들에 차이가 있음을 알 수 있습니다. 그래서 k-NN 분류를 배우면서 단 맛을 내게 하는 성분들의 비율을 가지고 그 과일이 무슨 과일인지 알 수 있을까 하는 의문이 생겼고 이 프로젝트를 진행하게 되었습니다.

2. 문제 정의

석류, 사과, 복숭아의 단맛을 내는 물질 중 글루코오스와 프락토오스의 비율을 독립변수로 하고, 과일의 종류를 종속변수로 하는 k-NN 모델을 실습해보았습니다.

II. 데이터 준비

① 석류, 사과, 복숭아 데이터 찾기

- ‘저설탕 및 고함량 올리고당 건강음료의 제조방법’ 연구의 실험 결과를 사용했습니다.

	글루코오스와 프락토오스의 비율				
석류	70.84	66.24	59.65	61.63	57.81
사과	44.43	43.21	24.90	27.18	25.91
복숭아	14.37	17.85	11.99	12.29	16.69

② 코딩에 맞게 데이터 변형

- 종속 변수 : 석류는 1, 사과는 2, 복숭아는 3 으로 지정했습니다.

Ⅲ. 구현 및 실험

① 패키지 불러오기

```
7 from sklearn.model_selection import train_test_split
8 from sklearn.neighbors import KNeighborsClassifier
9 from sklearn.metrics import accuracy_score
10 import matplotlib.pyplot as plt
11 import numpy as np
```

② 변수설정

- ‘글루코오스와 프락토오스의 비율’ 변수를 X로 설정하고 정의했습니다.

- X=[[70.84],[66.24],[59.65],[61.63],[57.81],[44.43],[43.21],[24.90],[27.18],[25.91],[14.37],[17.85],[11.99],[12.29],[16.69]]

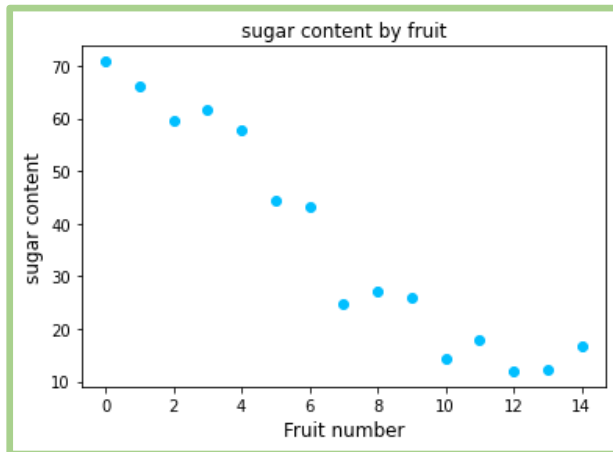
-과일은 각각 1,2,3 으로 정의해 Y 변수에 정의했습니다.

- Y=[1,1,1,1,1,2,2,2,2,2,3,3,3,3,3]

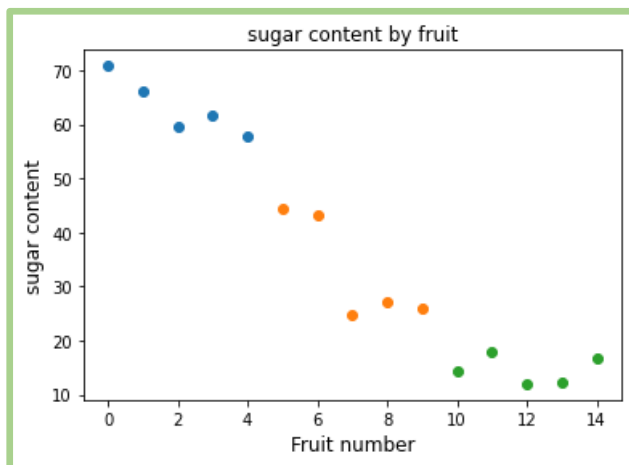
```
32 X=[[70.84],[66.24],[59.65],[61.63],[57.81],[44.43],[43.21],[24.90],[27.18],[25.91]
33 Y=[1,1,1,1,1,2,2,2,2,2,3,3,3,3,3]
```

③ 데이터 확인하기

k-NN 분류를 실습하기 전에 일단 데이터를 일단 먼저 확인했습니다.



15 개의 샘플을 가지고 분류를 진행할 것이므로 x 축은 15 개의 과일들의 번호이고, y 축은 그 번호의 과일 마다 가지는 ‘글루코오스와 프락토오스의 비율’ 입니다. 그리고 이 때,



파란 점이 석류, 주황색 점이 사과, 초록색이 복숭아입니다.

④ k-NN 분류

먼저 훈련데이터와 테스트할 데이터를 나눴습니다.

```
42 X_Train,X_Test,Y_Train,Y_Test=train_test_split(X,Y,test_size=0.2)
43
```

그 후 n_neighbors 의 값을 변경하면서 scores 를 확인해 봤습니다.

```
42 X_Train,X_Test,Y_Train,Y_Test=train_test_split(X,Y,test_size=0.2)
43
44 knn=KNeighborsClassifier(n_neighbors=5)
45 knn.fit(X_Train,Y_Train)
46
47 Y_Pred=knn.predict(X_Test)
```

```
In [70]: runfile('C:/Users/sin18/Desktop/fruit.py',
wdir='C:/Users/sin18/Desktop')
예측 : [3 3 1]
실제 : [3, 2, 1]
score : 0.6666666666666666
```

IV. 결과 분석

(1) n_neighbors=1 일 때

```
In [71]: runfile('C:/Users/sin18/Desktop/fruit.py',  
wdir='C:/Users/sin18/Desktop')  
예측 : [1 1 2]  
실제 : [1, 1, 2]  
score : 1.0
```

잘 예측이 되었음을 알 수 있습니다.

(2)n_neighbors=2 일 때

```
In [72]: runfile('C:/Users/sin18/Desktop/fruit.py',  
wdir='C:/Users/sin18/Desktop')  
예측 : [2 2 3]  
실제 : [2, 2, 3]  
score : 1.0
```

잘 예측이 되었음을 알 수 있습니다.

(2)n_neighbors 가 5 까지는 잘 예측이 되지만 6 이후부터는

```
In [76]: runfile('C:/Users/sin18/Desktop/fruit.py',  
wdir='C:/Users/sin18/Desktop')  
예측 : [2 2 2]  
실제 : [3, 2, 3]  
score : 0.3333333333333333
```

잘 예측이 되지 않음을 알 수 있습니다.

n_neighbors 는 5 가 가장 적절함을 알게 되었습니다.

V. 결론 도출

앞의 결과에서 맞힐 확률이 1 정도로 나온 것을 보아 k-NN 분류가 잘 이루어졌음을 알 수 있었습니다. 따라서 과일의 맛을 직접 보거나 먹지 않아도 성분만으로도 어떤 과일인지 예측할 수 있음을 알 수 있었다.

VI. 출처

[1] 설탕 및 고함량 올리고당 건강음료의 제조방법

[KR101195379B1 - 저설탕 및 고함량 올리고당 건강음료의 제조방법 - Google Patents](#)