인공지능

과제 제목: K-Nearest Neighbor With Iris Data

이름: 박수민

1. 개요

-K-Nearest Neighbor 알고리즘을 이용하여 Iris 꽃을 분류하는 과제

1. 구현 환경

-Anaconda, Jupyter

1. 알고리즘에 대한 설명

-K-NN 알고리즘은 머신러닝 알고리즘 중 지도학습에 속한다.

-시간 복잡도: O(n)

-거리 측정 방법: Euclidean distance

텍스트이(가) 표시된 사진

자동 생성된 설명

-Number of neighbors to consult: k

-Majority vote: k개의 가까운 데이터 중에서 가장 수가 많은 데이터를 major로 선정.

-Weighted majority vote: k개의 가까운 데이터 중에서 거리에 따르는 가중치를 합한 값에서 major를 선정.

-장점: 구현과 이해가 쉽다. Training이 빠르다, 정확도 높다

-단점: 계산양이 많다. O(n)만큼 저장공간. k값에 sensitive.

1. 데이터에 대한 설명
   1. Input Feature

* Data Size: 150
* Data Type: Float
* 4 features
* Sepal length(cm), Sepal width(cm), Petal length(cm), Petal width(cm)
  1. Target Output
* Data Type: String
* 3 class
* 0: Setosa, 1: Versicolor, 2: Virginica

1. 소스코드에 대한 설명

5.1

텍스트이(가) 표시된 사진

자동 생성된 설명

-테스트케이스와 나머지 데이터들과의 거리를 Euclidean방식으로 구현함

5.2

텍스트이(가) 표시된 사진

자동 생성된 설명

-k개의 이웃한 neighbor를 구한 뒤 sort\_target에 k개의 값들을 정렬함

5.3

텍스트이(가) 표시된 사진

자동 생성된 설명

-list\_target에는 k개의 이웃한 neighbor가 정렬되어있는데, counter함수와, most\_common함수를 이용해 가장 많은 값을 추출하는 majority vote를 구현함

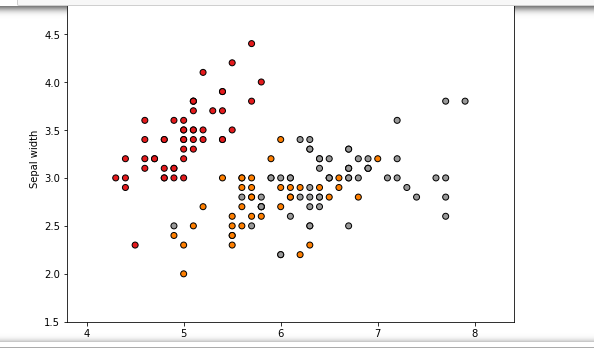
5.4

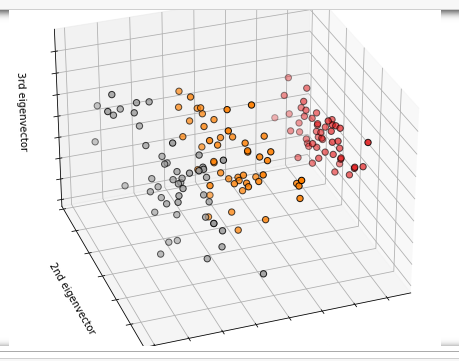
텍스트이(가) 표시된 사진

자동 생성된 설명

-list\_target에 정렬되어있는 순서가 가까운 거리부터 오름차순이므로 인덱스 작은 순서부터 가중치를 더한 뒤 가중치가 가장 높은 데이터를 찾을 수 있도록 하였음

1. 학습 과정에 대한 설명





-2차원과 3차원의 Iris 데이터를 보면 꽃의 각 종류별로 구역이 나눠져 있음을 알 수 있다. 테스트 케이스의 값과 나머지 값 들과의 거리 차이를 통하여 구분이 가능하다.

1. 결과 및 분석
   1. 각 테스트케이스와의 거리는 다음과 같이 계산되어진다.

텍스트, 신문이(가) 표시된 사진

자동 생성된 설명

* 1. 테스트 케이스와의 최소 거리 순으로 정렬한 인덱스를 출력한다. 여기서 0번째 인덱스는 테스트 케이스 자신이므로, 이후에 계산할 때는 1번째 인덱스부터 계산한다.

텍스트, 낱말맞추기게임, 영수증이(가) 표시된 사진

자동 생성된 설명

* 1. 테스트케이스와 가까운 k개의 인덱스를 출력한다(ex) k=5)



* 1. Weighted Majority Vote 방식에서는 다음 배열의 0,1,2 인덱스에 각각 꽃들의 가중치를 구현했다



7.5

K = 3일때, majority vote의 결과























K = 3일때, weighted majority vote의 결과























K = 5일때, majority vote의 결과























K = 5일때, weighted majority vote의 결과























K = 10일때, majority vote의 결과























K = 10일때, weighted majority vote의 결과





















