

HW 7

k - Nearest Neighbor 알고리즘을 이용해서 필기체 인식 시스템을 만들어보자.

아래와 같은 예제 데이터가 주어지며,
주어진 데이터로 학습하여 결과를 예측하고 돌려준다

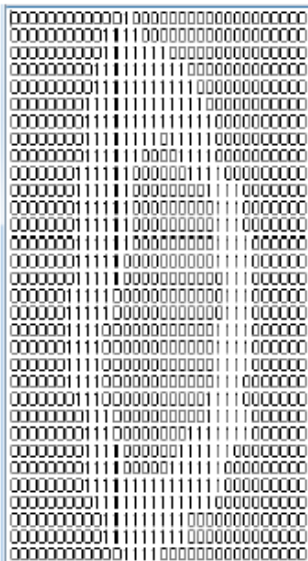
[illegible][illegible]

파일 이름에 데이터의 라벨이 있다.


제공하는 데이터

2. testDigits 폴더 : N_M.txt (N : 숫자, M : 데이터 ID)

이 폴더의 데이터들을 읽어서 숫자를 인식해야 한다.



kNN



영!!!

제공하는 데이터

2. testDigits 폴더 : N_M.txt (N : 숫자, M : 데이터 ID)

이 폴더의 데이터들을 읽어서 숫자를 인식해야 한다.

이 폴더의 모든 파일 데이터를 읽어서 숫자를 인식해야 한다.
그리고 인식한 결과와 실제 라벨을 확인해서 에러율을 계산하자.
cf) $\text{에러율} = \text{예측을 실패한 경우} / \text{전체 파일 개수} * 100$

제공하는 데이터

2. testDigits 폴더 : N_M.txt (N : 숫자, M : 데이터 ID)

이 폴더의 데이터들을 읽어서 숫자를 인식해야 한다.

이 폴더의 모든 파일 데이터를 읽어서 숫자를 인식해야 한다.
그리고 인식한 결과와 실제 라벨을 확인해서 에러율을 계산하자.

cf) $\text{에러율} = \text{예측을 실패한 경우} / \text{전체 파일 개수} * 100$

kNN 알고리즘에서 k를 변화시켜가면서 에러율을 출력하자.

K=1일 때는 xx%

K=2일 때는 yy%

...

HW7 cont'

제출물 : 소스코드

파일 이름 규칙 : Student<ID>.py

학번이 20151000이면 파일 이름은 Student20151000.py

실행 규칙 : python3 Student20151000.py <트레이닝 데이터 폴더> <학습 데이터 폴더>

Python3 Student20151000.py trainingDigits testDigits

여러분이 작성한 라이브러리는 별도 파일로 제출하지 않으며 하나의 파일로 제출한다.

github의 에 제출할 내용 및 기한

- 코드를 update하면 교수가 download하여 검사함.
- Github 상에서 HW4 폴더를 만들고 HW3 폴더에 코드 업로드

HW7 cont'

출력 규칙 : k가 1일 때부터 20일때까지 에러율을 차례대로 출력한다.
숫자 20개가 차례대로 한 줄에 한 숫자로 출력되며 소수점은 절사한다.

10

9

9

8

..

제출 마감 : 12월 27일 자정