**Machine Learning, Spring 2023**

**Homework 1**

Due 3/26 23:59

**코딩 숙제 제출 주의사항**

* **최종 제출물: 요구한대로 작성한 보고서와 코랩 노트북, Github repository 링크**
* **아주 Bb에 압축하지 않고 그대로 제출**
  + **워드파일의 경우, 워드파일 그대로 HW##\_학번\_이름.docx 로 제출. (ex HW10\_20XXXXXXX\_한지은)**
  + **코랩 노트북의 경우, HW##\_학번\_이름.ipynb 로 제출**
  + **코랩 노트북 파일을 업로드한 github repository 주소 링크**
* **주의사항**
  + **Github 업로드**
    - **github 의 last commit 을 기준으로 채점**
      * **따라서 bb에 업로드한 노트북 파일과 github repository에 올린 파일이 동일한 내용을 담고 있도록 할 것**
    - **github에 업로드된 코랩 파일은 실행 결과가 모두 출력되어 있는 상태로 업로드**
    - **github에 올린 노트북 파일에 Open in Colab 버튼이 있게 업로드 할 것 (코랩에서 github로 업로드 시 설정 가능)**
    - **github repository는 private 설정, collaborator로 조교 추가 완료되었는지 확인**
    - **요구사항을 지키지 않을 경우 각각 0.1점씩 감점 예정 (과제 하나당 1점으로 정규화 한 것을 가정)**
  + **보고서 제출**
    - **보고서에 코드 제출시 가독성에 유의하여 제출할 것**
    - **되도록이면 워드의 글자 크기와 유사하도록 코드를 캡처해서 올릴 것**
    - **주어진 코랩 노트북을 바탕으로 요구하는 코드를 구현 후 보고서에 해당 부분을 캡처**
    - **요구하는 코드만을 캡처하여 보고서에 제출**
    - **요구한 코드 이외에 다른 코드는 수정하지 말 것.**
  + **모든 요구한 코드에는 간단한 주석을 기입.**
* **과제가 많고 제출기간이 겹치는 경우가 있을 수 있으므로 제출 후 본인 과제가 정확히 제출되었는지 꼭 확인 요망.**
* **최종 점수는 비율적으로 수정해서 들어갈 예정**
* **딜레이 토큰의 경우 조교가 기록함.**

**코딩 숙제 관련 채점 방식 및 작성 유의 사항**

1. **서술하라고 하는 것은 워드에 직접 서술.** 
   1. **예시 문제 : 주어진 데이터의 특성을 파이썬을 이용하여 데이터를 읽은 후 각각의 변수마다 어떠한 타입인지 서술하시오.** 
      1. **답 : feature1 = 수치형 변수, feature2 = 범주형 변수 ..**
2. **코딩 문제의 경우 주어진 노트북에 기반하여 프로그래밍한다. 코딩을 해야 할 부분이 아닌 부분을 수정하여 과제를 해결해서는 안된다.** 
   1. **예시 문제) 중복값을 확인하고 중복값을 제거한 후 DataFrame의 차원을 확인하는 코드를 작성하여 캡처 후 보고서에 업로드 하세요.**

**Text, letter

Description automatically generated**

1. **채점할 시, 주피터 노트북의 경우 copy를 방지하고자 확인하는 용도이며(미제출시 채점 할 수 없음) 채점의 경우 보고서에 기반하여 채점함. 본인이 작성한 워드크기와 캡처한 코드의 글자가 유사하도록 보고서에 올릴 것.** 
   1. **Text

      Description automatically generated with medium confidence나쁜 예**

**과제 시작 전 유의사항**

**이번 학기 과제를 모두 동일한 repository(2023S-ML)에 업로드해주세요.**

**과제 시작**

**과제목표**

* **데이터의 형식에 대해서 이해하고 이를 변환할 수 있다.** 
  + **특히, numpy 와 pandas 라이브러리에 기반한 데이터 처리를 할 수 있다.**
* **결측값, 중복값, 이상치를 처리하고 다양한 시각화를 할 수 있다.**
* **기본적인 Exploratory Data Analysis를 수행할 수 있다.**
* **데이터는 동봉된 train.csv을 이용한다.**

**----------------------------------------------------------------------------------------------------------------------------------------**

**데이터 소개**

**대출 연체자 예측 데이터 in kaggle (https://www.kaggle.com/datasets/hemanthsai7/loandefault)**

**이 데이터셋은 약 10만 명의 대출 연체 관련 정보를 담고 있습니다. 여러 가지 정보들을 고려하여 대출 연체자를 예측하는 데이터셋입니다.**

**문제 유형 : Binary-class Classification 등**

**주요 데이터 변수**

**ID 고객 ID**

**Loan Amount 대출 금액**

**Funded Amount 투자 금액**

**Funded Amount Investor 투자자들이 승인한 대출 금액**

**Term 대출 기간 (월 단위)**

**Interest Rate 대출 이자(%)**

**Grade 은행에서 부여한 등급 (A ~ F)**

**Sub Grade 은행에서 부여한 하위 등급 (A ~ F)**

**Employment Duration MORTGAGE(담보), RENT(임차), OWN(자가)**

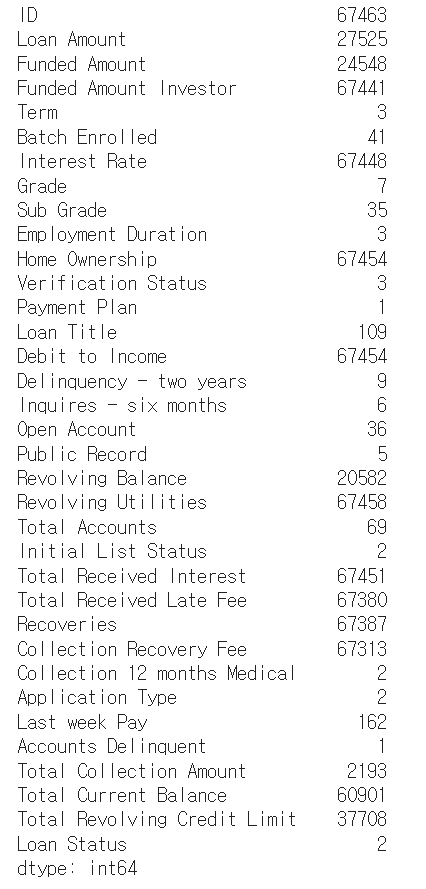
**Loan Status 대출 연체자 여부 [0 = 대출 연체자 아님, 1 = 대출 연체자]**

**문제 1**

**문제 1-1**

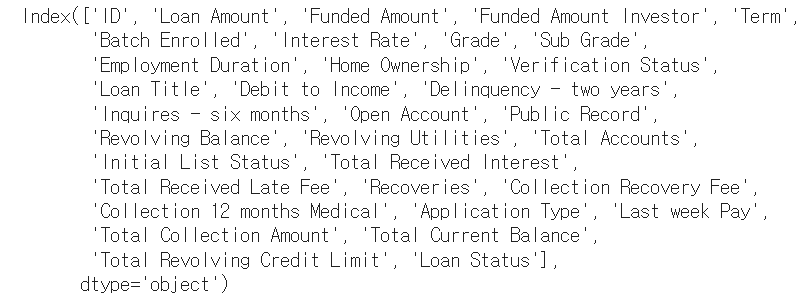
**DataFrame data의 각 feature에 포함된 unique value의 개수를 구하고 그 개수를 출력하여 보고서에 캡처하여 첨부하세요**

**(힌트: pandas에 있는 함수 사용)**



**문제 1-2**

**문제 1-1 결과를 바탕으로 고유값이 1인 column을 삭제하고 data.columns를 실행하여 그 결과를 보고서에 캡처하여 첨부하세요**



**문제 1-3**

**DataFrame data의 feature 이름에는 feature의 이름을 이루는 단어가 두 단어 이상인 경우, 공백이 포함되어 있습니다. feature 이름에 공백을 없애는 코드를 작성하고 data.columns를 실행하여 그 결과를 보고서에 첨부하세요**

텍스트, 편지이(가) 표시된 사진

자동 생성된 설명

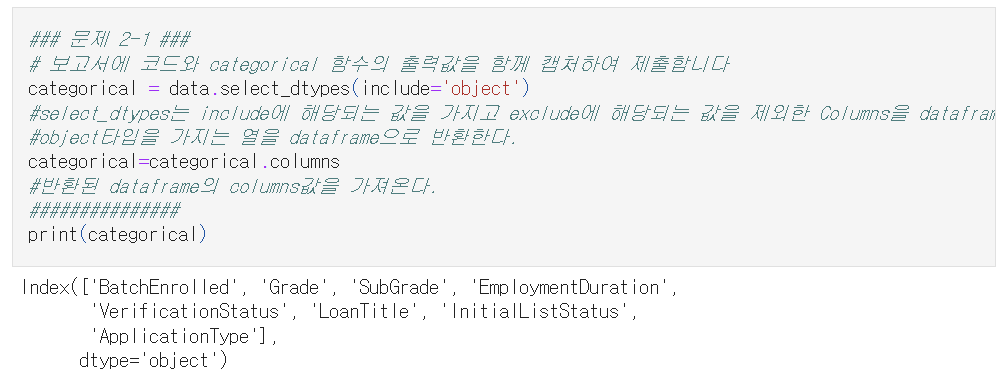
**문제 2**

**문제 2-1**

**위 ‘데이터 살펴보기’ 섹션에서 ‘data.dtypes’로 데이터의 type을 살펴보았습니다. ‘data.dtypes’ 결과를 참고하여 범주형 데이터(categorical data)를 가진 열의 이름을 담고 있는 List ```categorical```를 아래의 조건을 지켜서 만드세요**

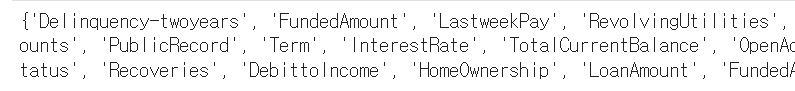
* **이 문제에서는 범주형 데이터가 ```object``` type을 가지고 있다고 가정합니다. (실제 데이터에서는 dtypes 결과가 int이더라도 categorical일 수 있습니다)**
* **categorical 함수의 data type은 Index가 아니라 List입니다.**

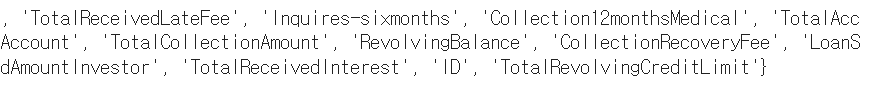
**보고서에 코드와 categorical 함수의 출력값을 함께 캡처하여 제출합니다.**



**문제 2-2**

**List categorical에 속하지 않은 feature들을 List numerical에 저장하고 numerical를 출력하여 보고서에 첨부하세요 (힌트: set(집합)**.

****

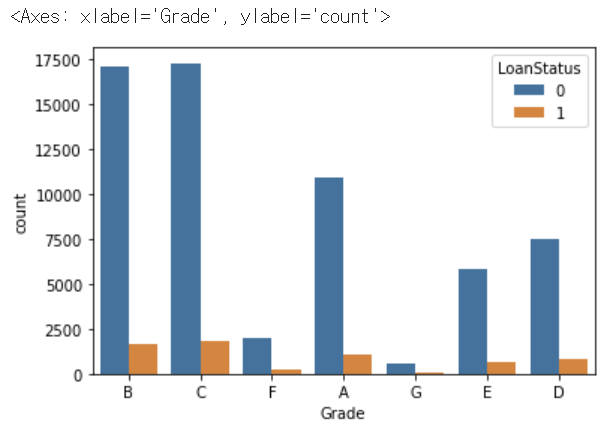


**문제 3 :: 데이터 시각화**

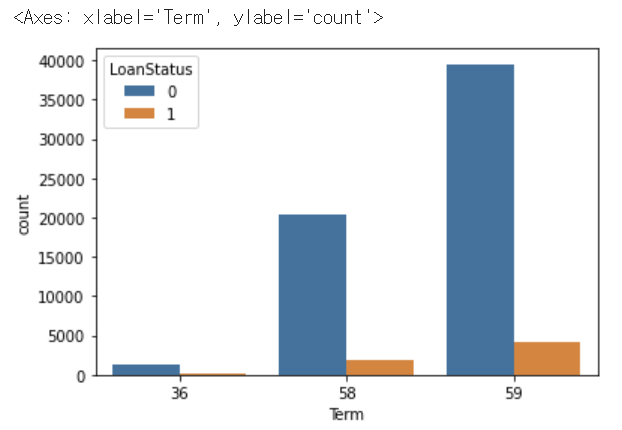
**문제 3-1**

**```seaborn.countplot```을 사용하여 각 Loan Status 에 대한 Grade, Term, Employment Duration의 countplot을 출력하여 3개의 countplot을 보고서에 캡처하여 첨부하세요.**

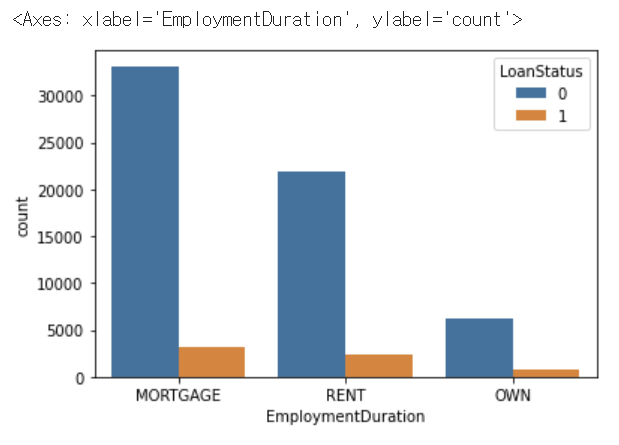
1. **Loan Status에 대한 Grade의 countplot**

****

1. **Loan Status에 대한 Term의 countplot**

****

1. **Loan Status에 대한 Employment Duration의 countplot**

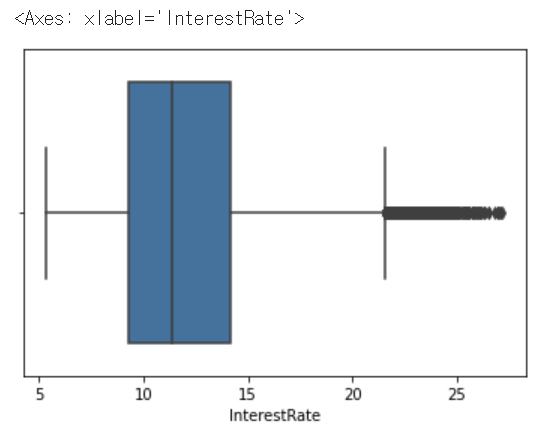
****

**문제 3-2**

**'Interest Rate'에 대한 boxplot을 그리고 캡처하여 보고서에 첨부하세요. 그리고 boxplot에 대해서 아래의 조건을 지켜 서술하세요. (강의노트 2, 40쪽 참고)**

* **서술에는 최소, 최대, 중앙값, Q1, Q3, outlier(이상치)를 포함합니다.**

1. **Boxplot 그림**

****

1. **Boxplot에 대한 설명**

**위의 그림은 InterRate에 대한 boxplot으로,**

**Rate의 최솟값은 5의 근처에 짧은 세로선으로 그려져 있고,**

**Q1(25%구간)은 파란박스의 제일 왼쪽 선에서 나타내어지고,**

**중앙값(Q2)는 Q1의 다음부분이자 10과 15의 사이 파란박스의 가운데 선이고,**

**Q3(75% 구간)은 15보다 아래 쪽에 파란박스의 오른쪽 선으로 나타나 있다,**

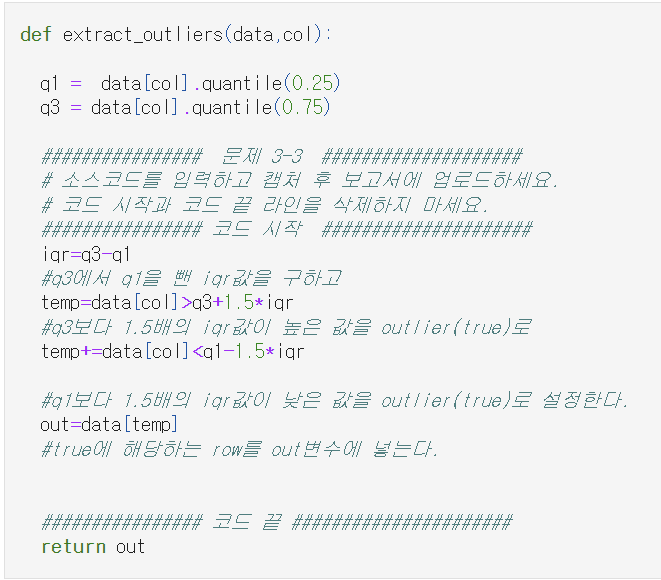
**최대값은 제일 오른쪽의 짧은 세로선이다. Q1-1.5\*IQR과 Q3+1.5\*IQR의 outlier(이상치)는 최대값을 넘어서 점으로 표시되어 있다.**

**차례대로 최소값, Q1, 중앙값(Q2), Q3, 최대값, 이상치 이다.**

**문제 3-3**

**문제 3-2에서 그린 boxplot에는 outlier가 있습니다. 'InterestRate'에 있는 outlier를 제거하는 함수 extract\_outliers() 의 빈 칸을 채워 outlier를 찾으세요. extract\_outliers 함수를 캡처하여 보고서에 첨부하세요.**

* **입력: 원본 데이터, outlier가 있는 열의 이름**
* **출력: outlier들만 있는 DataFrame**

****

**문제 4**

**문제1에서 구한 List categorical에는 'Grade' 와 'Sub Grade' feature가 포함되어있습니다. 이 두 가지 feature에 대해 살펴봅시다.**

**'Grade'와 'SubGrade'를 사용하여 두 categorical feature를 하나의 numerical feature로 통합한 'FinalGrade' 열을 생성합니다. 'FinalGrade' 열은 다음과 같은 규칙으로 생성됩니다.**

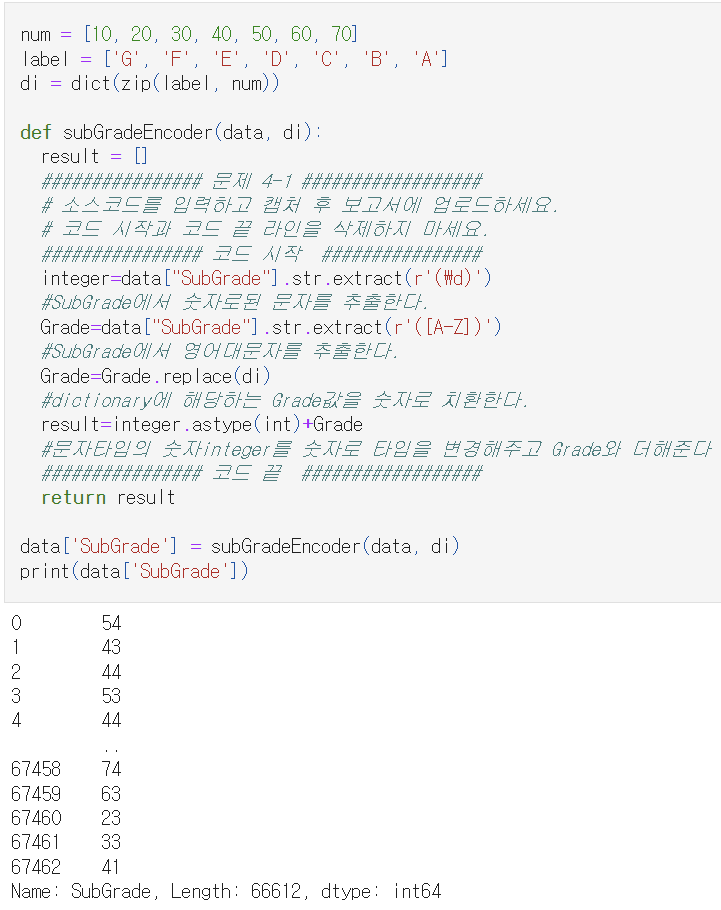
* **'Grade'는 A부터 G까지 7까지의 등급을 가지고 있습니다. 'Grade'가 A로 시작하면 700, B로 시작하면 600, ..., G로 시작하면 100을 부여합니다.**
* **'SubGrade'는 A부터 G까지 7까지의 등급을 가지고 있습니다. 'SubGrade'가 A로 시작하면 70, B로 시작하면 60, ..., G로 시작하면 10을 부여합니다. 'SubGrade'의 두 번째 자리(숫자 부분)를 앞에서 구한 정수값과 더한 값이 최종 'SubGrade'의 값입니다.**
* **'FinalGrade'는 각 'Grade'와 'SubGrade' 값을 더한 값입니다.**
* **예시) Grade: A, SubGrade: A4**

**A => 700, A4 => 70 + 4**

**Final Grade = 770**

**문제 4-1**

**주어진 list num과 label, di를 사용하여 'SubGrade' 값을 정수로 바꾸는 ```def subGradeEncoder```를 구현하고 코드와 셀 실행 결과를 모두 캡처하여 보고서에 첨부하세요.**

****

**문제 4-2**

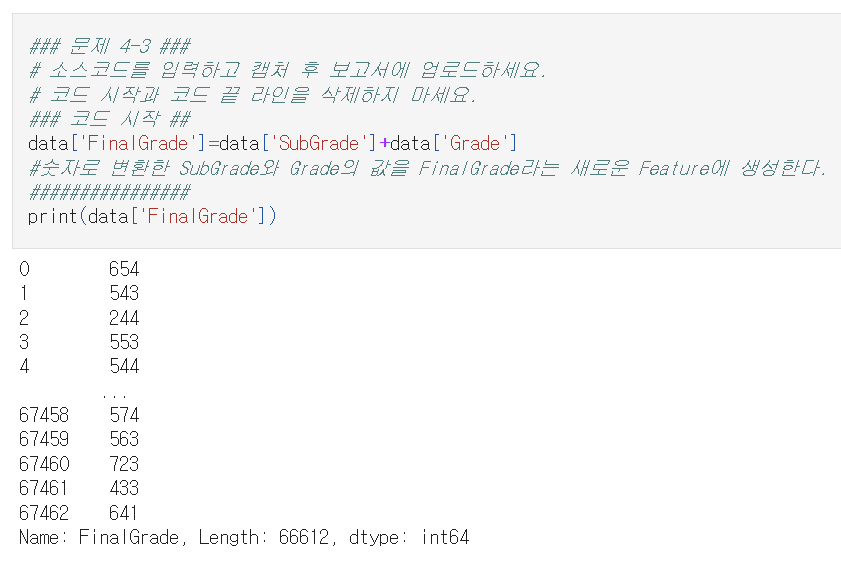
**주어진 list num과 label, di를 사용하여 'Grade' 값을 정수로 바꾸는 ```def gradeEncoder```를 구현하고 코드와 셀 실행 결과를 모두 캡처하여 보고서에 첨부하세요.**

**텍스트, 편지이(가) 표시된 사진

자동 생성된 설명**

**문제 4-3**

**'Sub Grade' 열의 값과 'Grade' 값을 합하여 최종적인 'Grade' 값을 담은 ```Final Grade```라는 열을 생성하고 코드와 셀 실행 결과를 모두 캡처하여 보고서에 첨부하세요.**

****

**문제 5**

=5인 -NN 알고리즘을 사용하여 하나의 test sample 을 분류하려고 합니다. 아래의 표에 training sample들 중에 와 가장 가까운 5개의 이웃과 각 training sample에서 까지의 거리가 주어져 있습니다.

|  |  |  |
| --- | --- | --- |
| Top 5 Neighbors | Class | =Distance to |
| 1 | C | 3 |
| 2 | B | 2 |
| 3 | C | 3 |
| 4 | A | 1 |
| 5 | C | 3 |

**문제 5-1**

위 결과에 Majority Voting을 적용하면 가 어떤 클래스로 분류되는 지를 쓰고 그 이유를 쓰세요. (이유를 쓰지 않으면 감점)

Majority Voting은 가중치가 없이 단순하게 가장 가까운 클래스 값을 따라간다

새로운 샘플와 가장 가까운 5개의 이웃에서 A, B, C 클래스의 개수는 순서대로 각각 1 ,1, 3개 이므로 새로운 샘플는 C로 분류된다.

**문제 5-2**

5-1에서 사용한 *Majority Voting*에서 각 vote에(각 이웃에) 가중치를 부여하는 방식을 적용하려고 합니다. 각 이웃 의 가중치는 거리 제곱의 역수로 for 정의됩니다.

* + 1. {A, B, C}에 있는 각 class label 에 대해서 score class label 에 속하는 이웃들의 가중치의 합으로 정의됩니다. score 를 계산하고 가중치를 고려하여 분류한 결과를 쓰세요.

**A== , B=, C= = =**

**{1, , } =(최소공배수 곱하기)=> {12,3,4} =(합만큼 나누기)=> {} 이므로, 가중치를 고려한 결과가 제일 높은 A로 분류된다.**

score **{}**

* + 1. 이 방식을 -NN에서 *weighted voting scheme*이라고 합니다. 이 *weighting scheme*의 영향과 *weighting scheme*을 (a)에서 사용한 *majority voting*과 비교했을 때 왜 더 좋은 방법이 될 수 있는지 서술하세요.

**KNN은 거리를 이용해서 가까운 이웃K개의 값 중 제일 빈도가 높게 등장한 것을 따라가는 방식이다.**

**해당알고리즘에서 ‘K개의 가까운 것’이라는 의미가 새로운 샘플의 결과값에 영향을 미치므로 해당 알고리즘은**

**거리에 민감하다고 볼 수 있다. 그러므로 가까운 것에는 큰 가중치를 두고, 먼 것에는 더 낮은 가중치를 줌으로써**

**가까운 이웃의 영향을 높일 수 있다.**

**다시 말해, KNN에서 majority voting을 했을 경우에는 근처에 있는 이웃의 수만이 새로운 샘플에 영향을 끼치고 노이즈에 민감한 반면, weighted voting을 했을 경우에는 이웃의 개수뿐 만 아니라 거리가 샘플의 클래스를 결정하는데 영향을 끼치므로 더 정확한 결과를 얻을 수 있다.**

**그리고 만약 A와 B클래스가 있을 경우, K의 수가 커져버리면 새로운 샘플의 분류 결과는 무조건 과반수를 따라가게 되는데, 이때도 가중치를 두어서 데이터가 편향되었을 때의 문제점을 완화할 수 있다.**

**깃허브 레포지토리 생성 후 그 레포지토리에 업로드 하기 <- 과제 점수에 포함**

**수고하셨습니다 :D**

**최종 제출물은 보고서, ipynb 파일, ipynb 파일을 업로드한 github repository 입니다.**

**제출하기 전에 첫 번째 페이지의 주의사항을 확인해주세요.**