

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**  
**KHAI THÁC DỮ LIỆU VÀ ỨNG DỤNG – CS313.N21**

---



**KHAI THÁC DỮ LIỆU GIÁO DỤC NHẪM DỰ ĐOÁN**  
**KẾT QUẢ HỌC TẬP CỦA SINH VIÊN**

**GVHD:** Nguyễn Thị Anh Thư  
Võ Tấn Khoa

**Sinh viên thực hiện:**

1	Nguyễn Phạm Hồng Duyên	20520477
2	Trần Thị Thu Hà	20521273
3	Nguyễn Thanh Thanh Trúc	20520829
4	Lê Văn Hùng	19520576
5	Lưu Quang Tiến Hoàng	20521342

## MỤC LỤC

1.	Giới thiệu .....	3
1.1	Tổng quan .....	3
1.2	Phát biểu bài toán.....	4
1.3	Thách thức .....	4
1.4	Đối tượng và phạm vi .....	5
1.5	Mục tiêu .....	5
2.	Mô hình giải bài toán.....	6
2.1	Các bước tiền xử lý dữ liệu.....	6
2.1.1	<i>Làm sạch dữ liệu</i> .....	6
2.1.2	<i>Tích hợp dữ liệu</i> .....	7
2.1.3	<i>Biến đổi dữ liệu</i> .....	10
2.1.4	<i>Rút gọn dữ liệu</i> .....	11
2.2	Các thuộc tính được sử dụng .....	12
2.3	Phương pháp đề xuất.....	15
3.	Cài đặt thực nghiệm.....	19
3.1	Dataset.....	19
3.2	Phương pháp đánh giá.....	19
3.2.1	<i>Confusion matrix</i> .....	20
3.2.2	<i>Accuracy</i> .....	21
3.2.3	<i>F1_score</i> .....	21
3.3	Phương pháp thực nghiệm .....	22
3.4	Kết quả thực nghiệm .....	23
4.	Kết luận và hướng phát triển .....	25
5.	Tài liệu tham khảo .....	26
6.	Phân công công việc.....	27

## NỘI DUNG

### 1. Giới thiệu

#### 1.1 Tổng quan

Đánh giá kết quả học tập của ứng viên trong quá trình học đại học là một trong những tiêu chí phổ biến được sử dụng trong quá trình tuyển dụng nguồn nhân lực của các công ty và doanh nghiệp. Trong giai đoạn đầu tiên của cuộc sống học đại học, sinh viên đối mặt với nhiều thay đổi và áp lực mới. Có rất nhiều yếu tố ảnh hưởng đến kết quả học tập của sinh viên, chẳng hạn điều kiện, khả năng học tập, môi trường học tập, độ tuổi, sức khỏe và tâm lý, giới tính hay nền tảng giáo dục trước đó. Liệu các yếu tố độ tuổi, giới tính và nền tảng giáo dục trước đó có thể ảnh hưởng nhiều đến sự thích nghi và thành công trong quá trình học tập của sinh viên.

Nhiều nghiên cứu đã được tiến hành để giúp phân tích sự ảnh hưởng của các yếu tố như nền tảng giáo dục bậc trung học phổ thông. Ngoài ra, một số nghiên cứu cũng đã nghiên cứu việc sử dụng kết quả học tập của các học kỳ trước đó để dự đoán kết quả học tập của học kỳ kế tiếp [1] [2] . Bài toán dự đoán sinh viên có qua một môn hay không dựa trên các môn cơ sở ngành là một nghiên cứu trong lĩnh vực khai thác dữ liệu giáo dục. [3] Mục đích của nghiên cứu nhằm giúp giảng viên và nhà trường đánh giá khả năng học tập của sinh viên và đưa ra các biện pháp hỗ trợ phù hợp để sinh viên có thể đạt được kết quả học tập tốt nhất.

Môn Lập trình hướng đối tượng là một môn học quan trọng trong chương trình đào tạo ngành công nghệ thông tin, đòi hỏi sinh viên phải nỗ lực học tập và nắm rõ nhiều kiến thức khó. Tình hình sinh viên rớt môn này là một vấn đề khá phổ biến trong ngành công nghệ thông tin. Theo thống kê, tỷ lệ sinh viên rớt môn này dao động từ 20% – 30%, tỷ lệ này có thể tùy thuộc vào chất lượng giảng dạy từ phía nhà trường, cũng như sự chuẩn bị và nỗ lực của sinh viên. Việc dự đoán sinh viên có qua môn Lập trình hướng đối tượng hay không nhằm đưa ra các giải pháp học tập và hỗ trợ học tập phù hợp, Các môn học cơ sở ngành như Giải tích, Xác suất thống kê, Nhập môn lập trình, Cấu trúc dữ liệu và giải thuật... sẽ được sử dụng để đánh giá. Đây là những môn học quan trọng trong chương trình đào tạo và có ảnh hưởng đến khả năng học tập của sinh viên

## 1.2 Phát biểu bài toán

Bài toán dự đoán sinh viên có qua một môn hay không dựa trên các môn cơ sở ngành đặt ra mục tiêu tìm hiểu và phân tích mối quan hệ giữa kết quả học tập của sinh viên trong các môn học trước đó và môn học dự đoán [3]. Khả năng dự đoán kết quả học tập có thể giúp các nhà quản lý giáo dục và giảng viên đưa ra các biện pháp hỗ trợ phù hợp để nâng cao hiệu quả học tập và thành công của sinh viên đại học.

Với mục đích dự đoán được dữ liệu của kết quả học các môn học trước như Giải tích, Xác suất thống kê, Nhập môn lập trình, Cấu trúc dữ liệu và giải thuật...có ảnh hưởng như thế nào đến việc sinh viên có qua môn Lập trình hướng đối tượng hay không, nhóm đề ra mục tiêu ban đầu xử lý lại bộ dữ liệu Education\_dataset\_v2 để được bộ dữ liệu phù hợp với mục đích của bài toán mà nhóm đã đặt ra. Sau đó tiến hành cài đặt và triển khai thực nghiệm các mô hình trên bộ dữ liệu sau khi đã được xử lý, từ đó thu thập được các kết quả thực nghiệm khác nhau trên từng mô hình, từng phương pháp.

Bài toán nhận đầu vào là điểm của sinh viên trong các môn học trước đó, đầu ra là dự đoán xem sinh viên có qua môn Lập trình hướng đối tượng hay không.

## 1.3 Thách thức

Kết quả học tập của sinh viên ảnh hưởng bởi nhiều yếu tố, trong đó độ tuổi, giới tính và nền tảng giáo dục trước đó là các yếu tố cần được xem xét kỹ lưỡng để đảm bảo sinh viên có một môi trường học tập tốt và đạt được kết quả tốt nhất. Bài toán gặp phải một số thách thức:

- Dữ liệu hiện tại có thể không đầy đủ, có giá trị thiếu. Để xây dựng mô hình dự đoán tốt, cần phải xử lý và tiền xử lý dữ liệu một cách cẩn thận để nâng cao độ chính xác của mô hình.
- Dữ liệu bị mất cân bằng giữa các lớp qua môn và không qua môn trong việc dự đoán xem sinh viên có qua một môn hay không dựa trên các môn cơ sở ngành.
- Hiện nay có nhiều thuật toán phân loại khác nhau và không phải thuật toán nào cũng phù hợp với bài toán đang quan tâm. Vì vậy, cần chọn thuật toán phù hợp và điều chỉnh các siêu tham số của thuật toán để đạt được kết quả tốt nhất.

- Việc đánh giá mô hình có thể mắc phải các lỗi như overfitting hoặc underfitting, do đó cần điều chỉnh mô hình và thực hiện lại quá trình đánh giá nhiều lần để đạt được kết quả tốt nhất.
- Khi có mô hình dự đoán tốt, cần giải thích kết quả của mô hình. Tuy nhiên, việc giải thích kết quả của một mô hình phức tạp có thể rất khó khăn và đòi hỏi kiến thức chuyên môn về thuật toán, xử lý dữ liệu và bài toán cụ thể.
- Bài toán dự đoán kết quả học tập của sinh viên có thể được áp dụng trong nhiều lĩnh vực khác nhau, từ giáo dục đến nhân sự. Tuy nhiên, để thực hiện áp dụng thực tế, cần đảm bảo tính khả thi của việc sử dụng dữ liệu và mô hình dự đoán trên các nền tảng và hệ thống khác nhau.

#### 1.4 Đối tượng và phạm vi

Bài toán dự đoán sinh viên có qua một môn hay không dựa trên các môn cơ sở ngành tập trung nghiên cứu phân tích mối quan hệ giữa kết quả các môn học cơ bản đến việc qua môn hay rớt môn học quan trọng của sinh viên ngành công nghệ thông tin. Nghiên cứu được thực hiện dựa trên bộ dữ liệu bao gồm các thông tin về kết quả học tập của sinh viên trong các môn học cơ sở ngành và kết quả học tập của sinh viên trong môn học Lập trình hướng đối tượng. Bộ dữ liệu được thu thập từ thông tin của các sinh viên theo học tại trường Đại học Công nghệ Thông tin – ĐHQG TPHCM từ khóa 8 (năm học 2013) đến khóa 14 (năm học 2019).

#### 1.5 Mục tiêu

Bài toán phân tích tác động của các yếu tố ảnh hưởng đến kết quả học tập của sinh viên hướng đến nhiều mục tiêu:

- Hỗ trợ quyết định trong tuyển sinh của các trường đại học, việc hiểu được tác động của độ tuổi, giới tính và nền tảng giáo dục trước đó đến kết quả học tập có thể giúp các trường đại học và nhà tuyển sinh đưa ra quyết định thông minh và công bằng.
- Giúp giảng viên và cố vấn học tập có một cái nhìn bao quát từ đó đưa ra các biện pháp hỗ trợ học tập phù hợp cho sinh viên, bao gồm các chương trình học tập cá nhân hóa, hướng dẫn chuyên sâu và các khóa đào tạo bổ sung.

- Tích hợp ứng dụng vào hệ thống quản lý học tập của trường để giúp giáo viên có thể dự đoán kết quả học tập của sinh viên và đưa ra các biện pháp nhằm giúp sinh viên cải thiện kết quả học tập.

## 2. Mô hình giải bài toán

### 2.1 Các bước tiền xử lý dữ liệu

#### 2.1.1 Làm sạch dữ liệu

Với mỗi tệp dữ liệu định dạng excel từ 15 tệp của bộ dữ liệu Education\_dataset\_V2.zip, lần lượt thực hiện các thao tác dưới đây để làm sạch bộ dữ liệu:

- Xóa các dòng dữ liệu chỉ chứa các giá trị NaN bằng hàm `pandas.dropna()`
- Tìm khóa chính của từng bảng dữ liệu, xóa dòng nếu khóa chính mang giá trị NaN
- Khử nhiễu dữ liệu bằng cách dùng hàm `pandas.unique()` liệt kê ra các giá trị duy nhất của từng cột dữ liệu, sau đó sử dụng hàm `pandas.replace()` để chuẩn hóa dữ liệu.
- Đối với một số cột trong bảng 04.xeploaiav, giá trị NaN được thể hiện dưới các dạng chuỗi string 'NULL)', 'NULL', dùng hàm `pandas.replace()` để đưa các chuỗi này về dạng `np.NaN`.

```
[ ] 1 df_12_baoluu['lydo'].unique()

array([' TN', ' ',
       ' Chứng chỉ Anh văn không đạt --> CNTN tốt nghiệp chuyển sang hệ CQĐT',
       ' QĐ điều chỉnh ngành TN', ' Tốt nghiệp'], dtype=object)

[ ] 1 # Chuyển ' ' thành nan
2 # Bỏ các khoảng trắng đầu chuỗi giá trị
3
4 df_12_baoluu.replace(' ', np.nan, inplace=True)
5 df_12_baoluu.replace(' TN', 'Tốt nghiệp', inplace=True)
6 df_12_baoluu.replace(' Chứng chỉ Anh văn không đạt --> CNTN tốt nghiệp chuyển sang hệ CQĐT',
7                       'Chứng chỉ Anh văn không đạt --> CNTN tốt nghiệp chuyển sang hệ CQĐT',
8                       inplace=True)
9 df_12_baoluu.replace(' QĐ điều chỉnh ngành TN', 'QĐ điều chỉnh ngành TN', inplace=True)
10 df_12_baoluu.replace(' Tốt nghiệp', 'Tốt nghiệp', inplace=True)
11
12 # Kiểm tra
13 df_12_baoluu['lydo'].unique()

array(['Tốt nghiệp', nan,
       'Chứng chỉ Anh văn không đạt --> CNTN tốt nghiệp chuyển sang hệ CQĐT',
       'QĐ điều chỉnh ngành TN'], dtype=object)
```

Hình 1. Ví dụ khử nhiễu dữ liệu ở cột lydo của bảng 12.baoluu bằng hàm `pandas.unique()` và `pandas.replace()`

Kết quả các file dữ liệu làm sạch được lưu trong thư mục edu\_data\_v2\_cleaned<sup>1</sup>.

### 2.1.2 Tích hợp dữ liệu

Kết hợp dữ liệu lop12\_matinh (từ bảng 05.thisinh), diachi\_tinhtp (từ bảng 01.sinhvien) và dữ liệu phân loại các khu vực dựa vào mã tỉnh và tên tỉnh từ website của đại học khoa học Thái Nguyên<sup>2</sup> tạo ra cột dữ liệu khu vực gồm 4 các giá trị ‘Khu vực 1’, ‘Khu vực 2’, ‘Khu vực 3’, ‘Khu vực 2 NT’. Tuy nhiên, đối với những thí sinh thuộc các diện tuyển thẳng và ưu tiên xét tuyển (1, 2, 3, 4, 5, 7) không sử dụng điểm thi THPTQG hoặc ĐGNL để xét tuyển thì không chứa thông tin lop12\_matinh hay lop12\_matruong thì join với bảng 01.sinhvien để sử dụng thông tin diachi\_tinhtp tham chiếu với dữ liệu các khu vực từ web. Dữ liệu từ cột diachi\_tinhtp đa dạng, gồm xóm, xã, quận, huyện, tỉnh/thành phố.

```
[11] 1 df3['diachi_tinhtp'].unique()

" 'Huyện Châu Thành'", " 'Huyện Hòa Thành'", " 'Xã An Hiệp'",
" 'Huyện Krông Ana'", " 'Huyện Long Thành'", " 'Huyện Ea Kar'",
" 'Huyện Đắk R\\'Lấp'", " 'Xã Vĩnh Hòa Hiệp'",
" 'Thành phố Biên Hòa'", " 'Thị xã Kiến Tường'", " 'KV8'",
" 'Huyện Đất Đỏ'", " 'Huyện Tân Thành'", " 'Tỉnh Đồng Tháp'",
" 'Xã Thành Thới A'", " 'Xã Bung Riềng'", " 'Huyện Đơn Dương'",
" 'Tỉnh Thanh Hoá'", " 'Tỉnh Thái Bình'", " 'Thị xã Ba Đồn'",
" 'Thành phố Hải Phòng'", " 'Huyện Krông Pắc'", " 'Quận Bình Tân'",
" 'Thành phố Rạch Giá'", " 'Thị xã Hà Tiên'", " 'Quận 6'",
" 'Thị trấn Núi Thành'", " 'Tỉnh Hải Dương'",
" 'Thành phố Thủ Dầu Một'", " 'Huyện Núi Thành'",
" 'Huyện Lấp Vò'", " 'Xã Hòa Định'", " 'Phường Tân Hiệp'",
" 'Huyện Hớn Quản'", " 'Quận Liên Chiểu'", " 'Thành phố Bến Tre'",
" 'Thị xã Đồng Xoài'", " 'Quận 7'", " 'Huyện Đạ Huoai'",
" 'Xã Long Giang'", " 'Quận Bình Thạnh'", " 'Thành phố Biên Hòa'",
" 'Huyện Trảng Bàng'", " 'Huyện Đông Hoà'", " 'Xã Ea Kuăng'",
" 'Thị Trấn Vĩnh Thuận'", " 'Xã Tân Ân'", " 'Huyện Di Linh'",
" 'Tỉnh Trà Vinh'", " 'Thành phố Hà Nội'", " 'Xã Long Mỹ'",
" 'Huyện Định Quán'", " 'Huyện Đức Trọng'", " 'Phường Linh Trung'",
" 'Xã Chi Lăng'", " 'Huyện Long Điền'", " 'Huyện Tuy Phước'",
" 'Huyện Phong Điền'", " 'Huyện Thoại Sơn'", " 'Xã Lạc An'",
" 'Thành phố Trà Vinh'", " 'Huyện Bình Sơn'", " 'Huyện Hải Lăng'"]
```

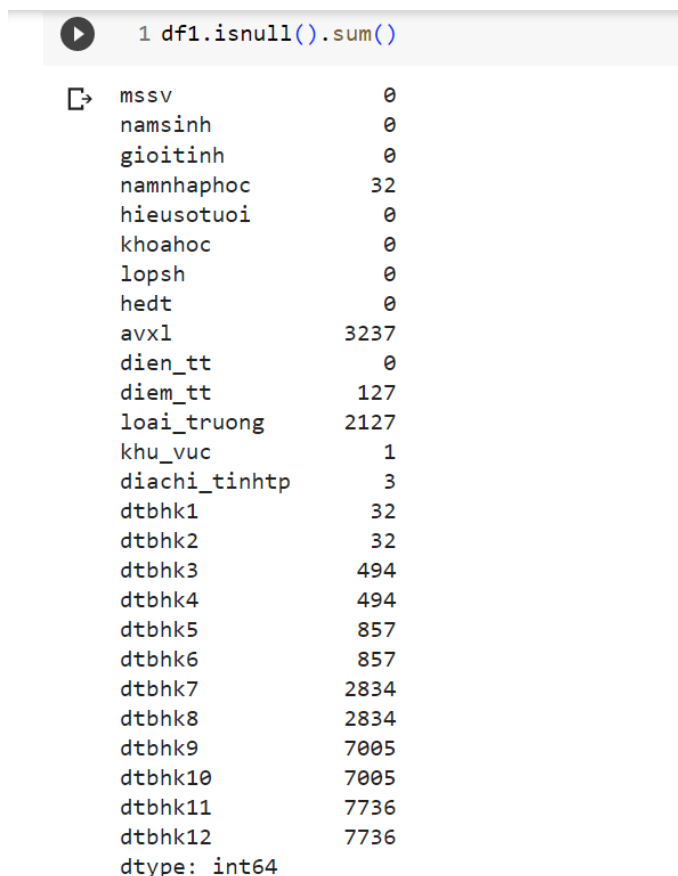
Hình 2. Ví dụ đa dạng các giá trị của cột diachi\_tinhtp của bảng 01.sinhvien

Sau khi khảo sát các thuộc tính của từng bảng với mục đích của bài toán, loại bỏ các thuộc tính dư thừa, không cần thiết, chỉ giữ lại một số thuộc tính cần thiết.

<sup>1</sup> <https://drive.google.com/drive/folders/1dOEmYiDf7oF6lodtfW4AW2zyxyG9sUdh?usp=sharing>

<sup>2</sup>

<https://tuyensinh.tnus.edu.vn/article/bang-phan chia khu vuc-2020-kv1-kv2-kv2nt-kv3>



```
1 df1.isnull().sum()
```

mssv	0
namsinh	0
gioitinh	0
namnhaphoc	32
hieusotuoi	0
khoahoc	0
lopsh	0
hedt	0
avx1	3237
dien_tt	0
diem_tt	127
loai_truong	2127
khu_vuc	1
diachi_tinhtp	3
dtbhk1	32
dtbhk2	32
dtbhk3	494
dtbhk4	494
dtbhk5	857
dtbhk6	857
dtbhk7	2834
dtbhk8	2834
dtbhk9	7005
dtbhk10	7005
dtbhk11	7736
dtbhk12	7736
dtype:	int64

Hình 3. Thống kê số lượng giá trị null của dữ liệu thông tin về nền tảng giáo dục và điểm trung bình từng học kỳ.

Áp dụng công thức xếp lớp anh văn từ file 00.ghichu.txt để điền các giá trị khuyết vào bảng dữ liệu:

*Công thức xếp lớp:*

2020: =IF(M6 < 250, "AVSC1", IF(M6 < 300, "AVSC2", IF(M6 < 350, "ENG01", IF(M6 < 400, "ENG02", IF(M6 < 450, "ENG03", IF(B6 = "CQUI", "Miễn ENG03", IF(M6 < 500, "ENG04", "ENG05"))))))))

2019: =IF(Q2 < 250, "AVSC1", IF(Q2 < 300, "AVSC2", IF(Q2 < 350, "ENG01", IF(Q2 < 400, "ENG02", IF(Q2 < 450, "ENG03", IF(F2 = "CQUI", "Miễn ENG03", IF(Q2 < 500, "ENG04", "ENG05"))))))))

2018:

MAMH      Điểm

AVSC1      <250

AVSC2      <300

ENG01      <350

ENG02      <400

ENG03      <450



ENG04 <500

ENG05 >=500

2017: =IF(E2>=96, "Anh văn 3", IF(E2>=72, "Anh văn 2", IF(E2>=40, "Anh văn 1", "Anh văn bổ túc")))

2016: =IF(E2<26, "Anh văn bổ túc", IF(E2<55, "Anh văn 1", "Anh văn 2"))

Các sinh viên thuộc khóa 8, 9, 10 không có điểm thi anh văn xếp lớp vì kì kiểm tra này chưa tổ chức vào những năm tuyển sinh của các khóa đó. Do vậy, dựa vào dữ liệu điểm học phần môn tiếng anh đầu tiên mà sinh viên đó học trích xuất từ bảng diem\_Thu để điền vào các chỗ dữ liệu còn bị khuyết. Hình ảnh dưới đây là các mã môn tiếng anh có trong bảng diem\_Thu:

```
1 df_02_diem_av['mamh'].value_counts()
2 ### Chú thích:
3 ## ENBT: Anh văn bổ túc (Mã môn cũ của BMAV --> nay là ENGBT của TTNN)
4 ## ADENG01: Tiếng anh tăng cường ~ ENG04 (Khoa HTTT)
5 ## ADENG02: Tiếng anh tăng cường ~ ENG04 (Khoa HTTT)
6 ## ADENG03: Tiếng anh tăng cường ~ ENG05 (Khoa HTTT)
7 ## ENGL1113: Tiếng anh I (Khoa HTTT) ~ ENG05
8 ## ENGL1213: Tiếng anh II (Khoa HTTT) ~ ENG06
9 ## EN001, EN002, EN003, EN004, EN005, EN006 ~ ENG01, ENG02, ENG03, ENG04, ENG05, ENG06
10 ## (Mã môn cũ của BMAV ~ Mã môn mới của TTNN)
```

EN001	2236
EN002	2015
EN004	1831
EN003	1557
ENBT	1083
EN005	1038
EN006	313
ENGL1113	135
ENGL1213	132
ADENG1	91
ADENG2	65
ADENG3	18

Name: mamh, dtype: int64

Hình 4. Các mã môn học tiếng anh có trong bảng diem\_Thu

Điền khuyết cột loại trường bằng giá trị 'THPT'. Kết quả sau khi điền khuyết thu được như hình dưới đây.

```
1 # Xem phân bố các loại trường sau khi điền khuyết
2 df['loaitruong'].value_counts()
```

THPT	6969
Chuyên	753
THCS&THPT	270
GDTX	52

Name: loaitruong, dtype: int64

Hình 5. Phân bố cột loaitruong sau khi điền khuyết dữ liệu

### 2.1.3 Biến đổi dữ liệu

Phân cấp khái niệm đối với các cột chứa điểm trung bình các kì (dtbhc1, dtbhc2, dtbhc3, dtbhc4, dtbhc5, dtbhc6, dtbhc7, dtbhc8, dtbhc9, dtbhc10, dtbhc11, dtbhc12) thành 4 loại Giỏi ( $8.0 \leq x \leq 10$ ), Khá ( $6.5 \leq x < 8.0$ ), Trung bình ( $5.0 \leq x < 6.5$ ) và Yếu ( $x < 5.0$ ).

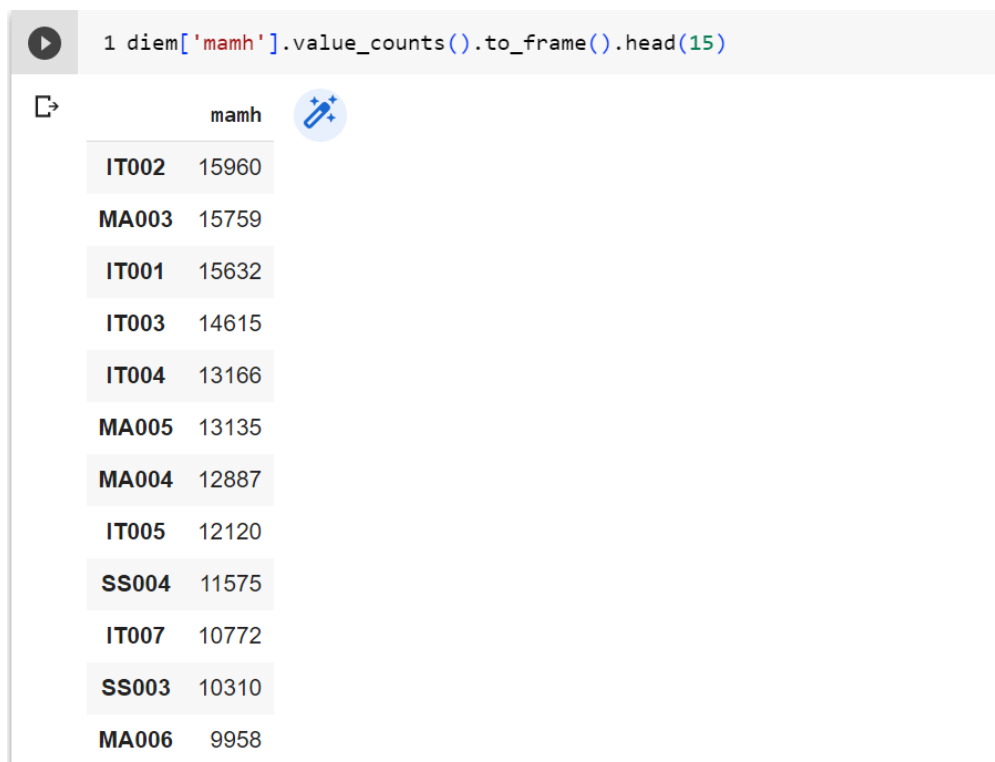
Cột xeploai\_diemtt (Xếp loại điểm trúng tuyển) được tạo ra từ cột diem\_tt, trong đó có 2 diện thí sinh trúng tuyển dựa vào điểm số là dientt=0 (Điểm thi THPTQG) và dientt=6 (Điểm thi ĐGNL). Từ đó, xếp loại theo các mức 1 (xuất sắc), 2 (giỏi), 3 (khá), 4 (trung bình). Đối với các thí sinh không có điểm thi do có dientt=1 (tuyển thẳng) thì được xếp vào mức 1. Các thí sinh có dientt thuộc [2, 3, 4, 5, 7] mà không có điểm thi thì được xếp vào mức 5 (khác).

1	msv	namsinh	giotinh	hieusotui	hoahoc	lopgh	hadt	diem_tt	loaitruong	huvuc	xeploai_diemtt	avd
2	48685401XPvAlbaE	1998	1	0	11	HTTT0001	COUI	0	THCS&THPT	Khu vực 3	3	ENGBT
3	F6059576XPvAlbaE	1998	0	0	11	HTTT0001	CTTT	0	THPT	Khu vực 3	4	ENG01
4	DC7421E3XPvAlbaE	1998	1	0	11	HTTT2016	COUI	0	THCS&THPT	Khu vực 2	3	ENG01
5	20D5A706XPvAlbaE	1998	1	0	11	MMTT0001	COUI	0	THPT	Khu vực 1	1	ENG01
6	2B06BF01XPvAlbaE	1998	1	0	11	MMTT0001	COUI	0	THPT	Khu vực 2	3	ENG01
7	7FA8CDB0XPvAlbaE	1998	1	0	11	PMCL2016.1	CLC	0	THPT	Khu vực 3	3	ENG01
8	B9D63D4DXPvAlbaE	1998	1	0	11	PMCL2016.1	CLC	0	THPT	Khu vực 1	3	ENG01
9	3A8C2FB6XPvAlbaE	1998	1	0	11	MMTT0001	COUI	0	Chuyên	Khu vực 2	3	ENG01
10	C34CB14FXPvAlbaE	1993	1	-5	11	MMTT0001	COUI	0	THPT	Khu vực 2 NT	3	ENG01
11	59AF4D97XPvAlbaE	1998	1	0	11	CNTT2016	COUI	0	THPT	Khu vực 3	2	ENG02
12	99203CEEXpVAlbaE	1998	1	0	11	HTTT0001	CTTT	0	THPT	Khu vực 2	4	ENG01
13	24500250XPvAlbaE	1998	1	0	11	KHMT0001	COUI	0	THPT	Khu vực 2	3	ENG01
14	7C663C97XPvAlbaE	1998	1	0	11	KTPM0001	COUI	0	THPT	Khu vực 2	2	ENG01
15	5DE27F64XPvAlbaE	1998	1	0	11	PMCL2016.1	CLC	0	THPT	Khu vực 3	3	ENG02
16	A34A5122XPvAlbaE	1998	1	-2	11	HTCL2016.1	CLC	0	THPT	Khu vực 2 NT	3	ENG01
17	152920E2XPvAlbaE	1998	1	0	11	HTCL2016.1	CLC	0	THPT	Khu vực 2 NT	4	ENG01
18	625D11A7XPvAlbaE	1998	1	0	11	KTPM0001	COUI	0	THPT	Khu vực 2	1	ENG02
19	D7158B5EXpVAlbaE	1998	0	0	11	MMTT0001	COUI	0	THPT	Khu vực 3	3	ENG01
20	46F21520XPvAlbaE	1998	0	0	11	KHMT0001	COUI	0	Chuyên	Khu vực 2	3	ENG01

Hình 6. 20 dòng đầu của dữ liệu 01\_sv\_nentangGD

1	msv	xeploai_hk1	xeploai_hk2	xeploai_hk3	xeploai_hk4	xeploai_hk5	xeploai_hk6	xeploai_hk7	xeploai_hk8	xeploai_hk9	xeploai_hk10	xeploai_hk11	xeploai_hk12
2	48685401XPvAlbaE	Trung bình khá	Trung bình	Trung bình khá	Trung bình	Kém	Trung bình	Trung bình khá	Yếu				
3	F6059576XPvAlbaE	Giỏi	Khá	Giỏi	Giỏi	Giỏi	Khá	Giỏi	Khá				
4	DC7421E3XPvAlbaE	Yếu	Kém	Yếu	Kém	Yếu	Kém						
5	20D5A706XPvAlbaE	Khá	Khá			Khá	Khá	Khá	Khá	Giỏi	Giỏi		
6	2B06BF01XPvAlbaE	Trung bình khá	Trung bình	Khá	Trung bình khá	Khá	Khá	Khá	Khá				
7	7FA8CDB0XPvAlbaE	Khá	Trung bình khá	Khá	Trung bình	Khá	Trung bình	Giỏi					
8	B9D63D4DXPvAlbaE	Khá	Trung bình	Trung bình	Trung bình khá	Yếu	Yếu	Khá	Khá				
9	3A8C2FB6XPvAlbaE	Khá	Trung bình	Trung bình khá	Trung bình khá	Trung bình	Trung bình khá	Trung bình khá	Trung bình khá			Giỏi	Kém
10	C34CB14FXPvAlbaE	Giỏi	Trung bình	Yếu	Kém	Trung bình	Kém	Trung bình	Kém	Khá	Yếu	Kém	Kém
11	59AF4D97XPvAlbaE	Trung bình	Trung bình khá	Khá	Khá	Khá	Khá	Trung bình khá	Khá				
12	99203CEEXpVAlbaE	Khá	Khá	Giỏi	Giỏi	Khá	Giỏi	Giỏi					
13	24500250XPvAlbaE	Khá	Trung bình khá	Khá	Trung bình	Trung bình	Kém	Kém	Kém	Kém	Kém	Yếu	Trung bình
14	7C663C97XPvAlbaE	Giỏi	Khá	Giỏi	Giỏi	Khá	Khá	Giỏi	Giỏi				
15	5DE27F64XPvAlbaE	Giỏi	Khá	Giỏi	Khá	Khá	Khá	Trung bình khá	Giỏi				
16	A34A5122XPvAlbaE	Trung bình	Yếu										
17	152920E2XPvAlbaE	Yếu	Kém										
18	625D11A7XPvAlbaE	Khá	Khá	Khá	Khá	Khá	Trung bình khá	Khá	Giỏi				
19	D7158B5EXpVAlbaE	Khá	Trung bình	Giỏi	Khá	Khá	Giỏi	Khá					
20	46F21520XPvAlbaE	Khá	Trung bình khá	Trung bình khá	Khá	Khá	Trung bình	Kém	Trung bình khá	Kém	Kém	Khá	Kém

Hình 7. 20 dòng đầu tiên của dữ liệu sv\_xeploaiHK



Hình 8. Danh sách các môn học có nhiều sinh viên học nhất sắp xếp theo thứ tự giảm dần

Các môn học có nhiều sinh viên đăng kí nhất là môn cơ sở nhóm ngành IT001 (Nhập môn lập trình), IT002 (Lập trình hướng đối tượng), IT003 (Cấu trúc dữ liệu và giải thuật), IT004 (Cơ sở dữ liệu), IT005 (Nhập môn mạng máy tính), IT007 (Hệ điều hành), MA003 (Đại số tuyến tính), MA004 (Cấu trúc rời rạc), MA005 (Xác suất thống kê), MA006 (Giải tích). Trong đó, môn IT002 có số lượng sinh viên trượt môn nhiều nhất nên chọn IT002 là biến dự đoán. Điểm được phân cấp về 4 mức đó là A ( $8.0 \leq x \leq 10$ ), B ( $6.5 \leq x < 8.0$ ), C ( $5.0 \leq x < 6.5$ ) và F ( $< 5.0$ ).

#### 2.1.4 Rút gọn dữ liệu

Rút gọn dữ liệu từ bảng diem\_Thu còn 11 gồm mssv và 10 môn học như đã chọn ở trên với điều kiện các sinh viên đều đăng kí học cả 10 môn. Chia dữ liệu thành hai bảng diem\_v1 và diem\_v2 để dự đoán điểm theo bài toán hồi quy và dự đoán khả năng qua môn hoặc không qua môn IT002 dựa vào kết quả của 9 môn còn lại.

	mssv	IT001	IT003	IT004	IT005	IT007	MA003	MA004	MA005	MA006	IT002
0	000AD0D8XPvAibaEXe+RQyZpP6sq6qqIPZXybx3Q	8.2	8.9	9.2	7.9	8.3	9.1	9.8	8.9	9.4	7.8
1	000C4D39XPvAibaEXe8AohUVcWybLdGDVP9n1cLa	9.1	8.2	7.6	7.5	7.9	8.8	9.3	7.8	10.0	9.2
2	0013D6E5XPvAibaEXe85hkLGAJy3XgK9pA18A31	8.5	8.0	7.1	6.9	8.3	8.2	8.4	8.8	9.2	8.3
3	0018C59CXPvAibaEXe8C3lhl2dNniH+SYgLosUA	9.1	8.4	7.7	6.5	7.0	7.5	8.2	9.4	6.8	7.7
4	001E045BXPvAibaEXe+n07P56kWx2N6EoOCUJBA4	7.1	8.0	6.7	5.9	6.8	8.0	6.7	5.5	6.9	5.8
...	...	...	...	...	...	...	...	...	...	...	...
5645	FFC5E1C7XPvAibaEXe915NmCViXOMEuHDtaXvcWR	5.2	5.6	5.1	6.3	7.1	5.5	6.9	6.9	6.2	6.9
5646	FFDAA424XPvAibaEXe9eTi/C0qygJA9JvUYHv+ci	8.4	7.3	7.3	7.0	7.1	8.2	8.7	6.1	8.1	7.0
5647	FFDC81D5XPvAibaEXe9/w7qCPr4fx1roEzeCxcg8	9.4	9.0	8.8	7.4	8.3	9.4	9.5	9.5	9.4	7.8
5648	FFE53E27XPvAibaEXe+boSxJoV2lkIPm7Byt5HdS	9.3	8.1	7.5	7.0	7.7	9.5	6.6	10.0	9.8	7.7
5649	FFEF294AXPvAibaEXe/ceziXFRXnLc/x/K0hVw4d	6.0	5.9	7.2	7.9	5.5	8.2	8.1	7.3	8.7	6.3

5650 rows × 11 columns

Hình 9. Mô tả dữ liệu 03\_diem\_v1

	mssv	IT001	IT003	IT004	IT005	IT007	MA003	MA004	MA005	MA006	IT002
0	000AD0D8XPvAibaEXe+RQyZpP6sq6qqIPZXybx3Q	A	A	A	B	A	A	A	A	A	1
1	000C4D39XPvAibaEXe8AohUVcWybLdGDVP9n1cLa	A	A	B	B	B	A	A	B	A	1
2	0013D6E5XPvAibaEXe85hkLGAJy3XgK9pA18A31	A	A	B	B	A	A	A	A	A	1
3	0018C59CXPvAibaEXe8C3lhl2dNniH+SYgLosUA	A	A	B	B	B	B	A	A	B	1
4	001E045BXPvAibaEXe+n07P56kWx2N6EoOCUJBA4	B	A	B	C	B	A	B	C	B	1
...	...	...	...	...	...	...	...	...	...	...	...
5645	FFC5E1C7XPvAibaEXe915NmCViXOMEuHDtaXvcWR	C	C	C	C	B	C	B	B	C	1
5646	FFDAA424XPvAibaEXe9eTi/C0qygJA9JvUYHv+ci	A	B	B	B	B	A	A	C	A	1
5647	FFDC81D5XPvAibaEXe9/w7qCPr4fx1roEzeCxcg8	A	A	A	B	A	A	A	A	A	1
5648	FFE53E27XPvAibaEXe+boSxJoV2lkIPm7Byt5HdS	A	A	B	B	B	A	B	A	A	1
5649	FFEF294AXPvAibaEXe/ceziXFRXnLc/x/K0hVw4d	C	C	B	B	C	A	A	B	A	1

5650 rows × 11 columns

Hình 10. Mô tả dữ liệu 04\_diem\_v2

## 2.2 Các thuộc tính được sử dụng

### – 01\_sv\_nentangGD.csv

01\_sv\_nentangGD gồm 8044 dòng và 12 cột, trong đó, các thuộc tính được sử dụng làm input của mô hình dự đoán mức độ ảnh hưởng của độ tuổi, giới tính và nền tảng giáo dục lên kết quả học kì 1:

- **gioitinh** (Giới tính) (int): có 2 giá trị 1 (nam) và 0 (nữ)
- **hieusotuoi** (Hiệu số tuổi) (int): hiệu số của năm sinh của sinh viên với năm tương ứng với một sinh viên nhập học năm 18 tuổi (ngay sau khi kết thúc 12 năm học phổ thông)

Vd: sinh viên sinh năm 2000 có khóa học là 14

$hieusotuai = 2000 - 2001 = -1$

(vì sinh viên khi nhập học khóa 14 năm 18 tuổi sinh năm 2001)

- **dientt** (diện trúng tuyển) (int): Bao gồm các giá trị 'THPT': 0, 'TT-Bộ': 1, 'CUTUYEN': 2, 'U'T-Bộ': 3, '30A': 4, 'U'T-ĐHQG': 5, 'ĐGNL': 6, 'CCQT': 7.
- **loaitruong** (loại trường THPT) (object): phân loại trường THPT của sinh viên thuộc 1 trong 4 loại sau đây: 'THCS&THPT', 'THPT', 'Chuyên', 'GDTX'.
- **khuvuc** (khu vực) (object): Thông tin địa phương của sinh viên thuộc khu vực nào trong 4 khu vực: 'Khu vực 1', 'Khu vực 2', 'Khu vực 3', 'Khu vực 2 NT'.
- **xeploai\_dientt** (Xếp loại điểm trúng tuyển) (int): Có 2 diện thí sinh trúng tuyển dựa vào điểm số là  $dientt=0$  (Điểm thi THPTQG) và  $dientt=6$  (Điểm thi ĐGNL). Từ đó, xếp loại theo các mức 1 (xuất sắc), 2 (giỏi), 3 (khá), 4 (trung bình). Đối với các thí sinh không có điểm thi do có  $dientt=1$  (tuyển thẳng) thì được xếp vào mức 1. Các thí sinh có  $dientt$  thuộc [2, 3, 4, 5, 7] mà không có điểm thi thì được xếp vào mức 5 (khác).
- **avx1** (Kết quả anh văn xếp lớp đầu vào) (object): kết quả kì thi xếp lớp anh văn đầu vào. Gồm ENGBT (Anh văn bổ túc), ENG01 (Anh văn 1), ENG02 (Anh văn 2), ENG03 (Anh văn 3), ENG04 (Anh văn 4), ENG05 (Anh văn 5), PASSENG03 (miễn anh văn 3 đối với sinh viên hệ đại trà).

#### – 02\_sv\_xeploaiHK.csv

02\_sv\_xeploaiHK gồm 8044 dòng và 12 cột, trong đó, kết quả học tập của các kì trước đó sẽ làm input để dự đoán kết quả của kì này. Ví dụ muốn dự đoán kết quả của học kì 4, input là xeploai\_hk1, xeploai\_hk2, xeploai\_hk3 và output là xeploai\_hk4.

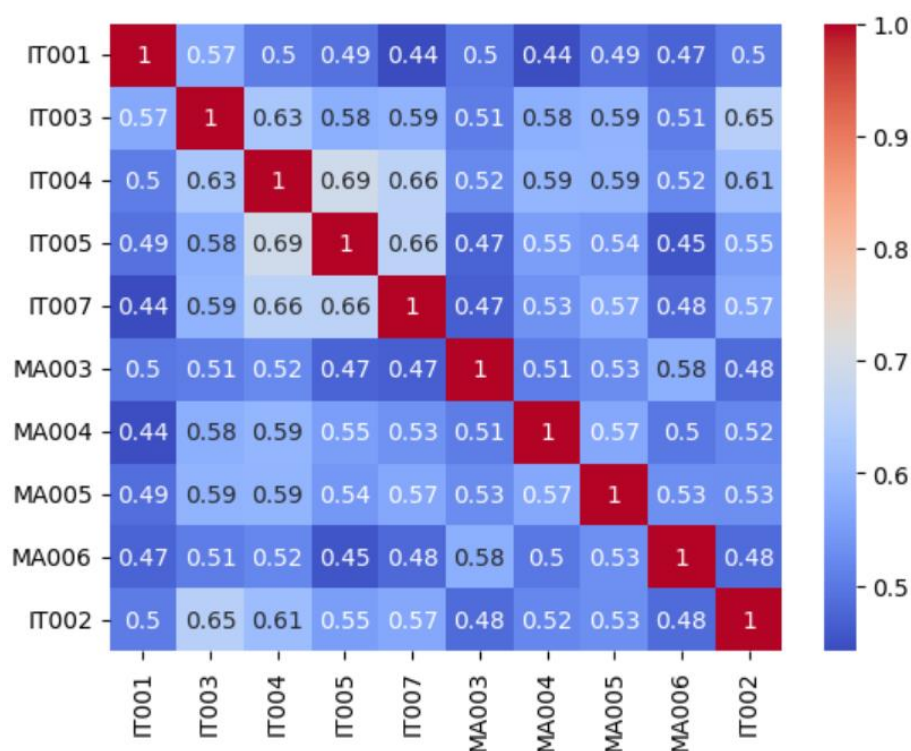
- Các cột xeploai\_hk1, xeploai\_hk2, xeploai\_hk3, xeploai\_hk4, xeploai\_hk5, xeploai\_hk6, xeploai\_hk7, xeploai\_hk8, xeploai\_hk9, xeploai\_hk10, xeploai\_hk11, xeploai\_hk12 (object): Là kết quả xếp loại học lực của các kì học tương ứng (thời gian đào tạo tối đa là 6 năm 12 học kì) thuộc 1 trong 4 loại (Giỏi,

Khá, Trung bình, Yếu) đã được phân cấp khái niệm như trình bày ở mục 2.1.3. Khi nào sinh viên không học thì có giá trị NaN.

– **03\_diem\_v1.csv**

03\_diem\_v1 bao gồm 5650 dòng và 11 cột, một cột mssv và 10 cột còn lại tương ứng với 10 môn học với số lượng sinh viên đông nhất đã được chọn ở bước tiền xử lý dữ liệu, trong đó IT002 là biến mục tiêu cần dự đoán dựa trên 9 cột tương ứng với 9 môn học còn lại.

- IT001, IT002, IT003, IT004, IT005, IT007, MA003, MA004, MA005, MA006 (float): điểm trung bình học phần của các môn học tương ứng là IT001 (Nhập môn lập trình), IT002 (Lập trình hướng đối tượng), IT003 (Cấu trúc dữ liệu và giải thuật), IT004 (Cơ sở dữ liệu), IT005 (Nhập môn mạng máy tính), IT007 (Hệ điều hành), MA003 (Đại số tuyến tính), MA004 (Cấu trúc rời rạc), MA005 (Xác suất thống kê), MA006 (Giải tích).



Hình 11. Heatmap thể hiện mối tương quan giữa các biến liên tục trong 03\_diem\_v1

– **04\_diem\_v2.csv**

04\_diem\_v2 bao gồm 5650 dòng và 11 cột, một cột mssv và 10 cột còn lại tương ứng với 10 môn học với số lượng sinh viên đông nhất đã được chọn ở bước tiền xử lý dữ liệu, trong đó IT002 là biến mục tiêu cần dự đoán dựa trên 9 cột tương ứng

với 9 môn học còn lại. 9 cột đó có điểm số kiểu float được phân cấp thành 4 loại A, B, C, F và biến mục tiêu IT002 được phân cấp thành 2 loại là 1 (qua môn,  $\geq 5.0$ ) và 0 (trượt môn,  $< 5$ ).

- IT001, IT003, IT004, IT005, IT007, MA003, MA004, MA005, MA006 (object): xếp loại điểm trung bình học phần của các môn học tương ứng là IT001 (Nhập môn lập trình), IT003 (Cấu trúc dữ liệu và giải thuật), IT004 (Cơ sở dữ liệu), IT005 (Nhập môn mạng máy tính), IT007 (Hệ điều hành), MA003 (Đại số tuyến tính), MA004 (Cấu trúc rời rạc), MA005 (Xác suất thống kê), MA006 (giải tích). Có 4 loại đó là A ( $8.0 \leq x \leq 10$ ), B ( $6.5 \leq x < 8.0$ ), C ( $5.0 \leq x < 6.5$ ) và F ( $< 5.0$ ).
- IT002 (int): kết quả qua môn hoặc trượt môn IT002 (Lập trình hướng đối tượng). Có 2 giá trị 1 (qua môn) và 0 (trượt môn).

### 2.3 Phương pháp đề xuất

Để xây dựng đề tài này, chúng tôi đã sử dụng nhiều mô hình máy học, độ đo và các phương pháp như sau:

#### – Support Vector Machine (SVM):

SVM là một phương pháp phân loại được sử dụng để dự đoán điểm của sinh viên trong các học kỳ. SVM tạo ra một siêu mặt phẳng tối ưu để phân chia các điểm dữ liệu thành hai lớp dựa trên các đặc trưng của sinh viên. SVM có thể xử lý dữ liệu số và dữ liệu rời rạc, và có khả năng tốt trong việc xử lý dữ liệu nhiễu và dữ liệu có số chiều cao.

#### – Logistic Regression:

Logistic Regression là một phương pháp hồi quy được sử dụng để dự đoán điểm của sinh viên trong các học kỳ. Logistic Regression ước lượng xác suất để một sinh viên thuộc vào một lớp cụ thể. Logistic Regression là mô hình tuyến tính nhưng có thể được mở rộng để xử lý dữ liệu phi tuyến tính thông qua các phép biến đổi đặc trưng.

#### – Random Forest:

Random Forest là một mô hình ensemble dựa trên cây quyết định, có thể được sử dụng để dự đoán điểm của sinh viên trong các học kỳ. Random Forest bao gồm

một tập hợp các cây quyết định độc lập, được huấn luyện trên các mẫu dữ liệu con ngẫu nhiên từ tập huấn luyện. Kết quả dự đoán cuối cùng được tính bằng cách áp dụng phương pháp voting hoặc trung bình kết quả từ các cây quyết định.

– **K-Nearest Neighbors (KNN):**

KNN là một phương pháp dự đoán dựa trên việc so sánh với các điểm gần nhất, có thể được sử dụng để dự đoán điểm của sinh viên trong các học kỳ. KNN dựa trên nguyên tắc rằng các sinh viên có điểm tương tự sẽ có đặc trưng gần nhau trong không gian đặc trưng. KNN xác định lớp của một sinh viên mới bằng cách so sánh nó với các sinh viên xung quanh dựa trên khoảng cách Euclid.

– **Decision Tree:**

Decision Tree là một phương pháp dự đoán dựa trên cây quyết định, có thể được sử dụng để dự đoán điểm của sinh viên trong các học kỳ. Decision Tree xây dựng cây quyết định dựa trên các quy tắc phân chia dữ liệu, để tìm ra các đặc trưng quan trọng trong việc dự đoán điểm. Decision Tree có khả năng xử lý dữ liệu số và dữ liệu rời rạc, cung cấp diễn giải cho quyết định dựa trên cây.

– **Sử dụng kỹ thuật Ensemble Learning (Voting Classifier)**

Bằng cách kết hợp các mô hình khác nhau, có thể tăng khả năng tổng quát hóa và cải thiện hiệu suất dự đoán.

Sử dụng phương pháp tối ưu hóa tham số mô hình:

- **KNeighborsClassifier:** Mô hình phân loại dựa trên K-Nearest Neighbors (KNN).

**n\_neighbors:** Số lượng láng giềng gần nhất sẽ được sử dụng để đưa ra dự đoán.

- **DecisionTreeClassifier:** Mô hình cây quyết định.

**max\_depth:** Độ sâu tối đa của cây quyết định.

- **SVC:** Mô hình Support Vector Machine (SVM).

**kernel:** Hạt nhân (kernel) được sử dụng trong SVM.



C: Tham số điều chỉnh độ mềm của SVM.

- LogisticRegression: Mô hình hồi quy logistic.

solver: Phương pháp tối ưu hóa trong quá trình huấn luyện mô hình logistic regression.

- RandomForestClassifier: Mô hình Random Forest.

n\_estimators: Số lượng cây trong ensemble.

max\_depth: Độ sâu tối đa của các cây trong ensemble.

- VotingClassifier: Bộ Ensemble sử dụng phương pháp Voting.

estimators: Danh sách các mô hình và tên tương ứng của chúng.

voting: Phương pháp tính toán kết quả cuối cùng của Ensemble.

#### – SMOTE (Synthetic Minority Over-sampling Technique)

Bằng cách áp dụng phương pháp SMOTE trong quá trình chuẩn bị dữ liệu cho việc huấn luyện mô hình, sẽ giúp cho bộ dữ liệu giải quyết được vấn đề mất cân bằng về số lượng các mẫu.

Mục đích của việc sử dụng SMOTE trong huấn luyện mô hình máy học là giúp cân bằng dữ liệu giữa các nhóm có số lượng mẫu khác nhau. Khi một tập dữ liệu có sự mất cân bằng lớn giữa các nhóm, các mô hình học máy có thể dễ dàng bị thiên vị một nhóm có số lượng mẫu lớn hơn. Do đó, việc tạo ra các mẫu nhân tạo thông qua SMOTE có thể giúp cân bằng lại tỉ lệ giữa các nhóm, từ đó cải thiện hiệu suất của các mô hình học máy được huấn luyện bằng dữ liệu đã được xử lý SMOTE. [4]

#### – Stochastic Gradient Descent (SGD):

Một biến thể của thuật toán Gradient Descent. Trong Gradient Descent, có một thuật ngữ gọi là “batch”, biểu thị tổng số mẫu từ tập dữ liệu được sử dụng để tính toán gradient cho mỗi lần lặp. Trong tối ưu hóa Gradient Descent điển hình, như Batch Gradient Descent, batch được coi là toàn bộ tập dữ liệu. Tuy nhiên, khi tập dữ liệu của chúng ta lớn hơn, việc sử dụng toàn bộ tập dữ liệu trở nên rất tốn kém về mặt tính toán để thực hiện. Vấn đề này được giải quyết bằng Stochastic Gradient Descent. Trong SGD, nó chỉ

sử dụng một mẫu duy nhất, tức là kích thước batch của một mẫu, để thực hiện mỗi lần lặp. Mẫu được xáo trộn ngẫu nhiên và được chọn để thực hiện lặp lại. [5]

– **MLP là viết tắt của Multi-Layer Perceptron (Perceptron đa tầng):**

Một loại mạng nơ-ron nhân tạo (Artificial Neural Network – ANN) feedforward. Nó bao gồm nhiều lớp các nút nơ-ron được kết nối đầy đủ với nhau. Mỗi nút nơ-ron trong một lớp được kết nối với tất cả các nút nơ-ron trong lớp tiếp theo. [6]

– **Passive Aggressive:**

Một thuật toán phân loại trực tuyến (online classification) có thể được sử dụng để giải quyết các bài toán phân loại nhị phân và đa lớp. Nó hoạt động bằng cách cập nhật mô hình chỉ khi mô hình đưa ra dự đoán sai.

Thuật toán Passive Aggressive có hai biến thể chính: PA và PA-I. PA cập nhật trọng số của mô hình bằng cách giải quyết một bài toán tối ưu hóa có ràng buộc. PA-I thêm một tham số điều chuẩn vào bài toán tối ưu hóa để kiểm soát độ lớn của bước cập nhật.

Passive Aggressive thuộc nhóm các thuật toán máy vector hỗ trợ trực tuyến (online support vector machines) và có thể được sử dụng để giải quyết các bài toán phân loại với dữ liệu dòng chảy (streaming data) hoặc dữ liệu quá lớn để xử lý một lần. [7]

– **Mini Batch KMeans:**

Một biến thể của thuật toán KMeans, được sử dụng để phân cụm dữ liệu. Nó hoạt động tương tự như KMeans nhưng sử dụng các lô (batch) dữ liệu nhỏ hơn để tăng tốc độ huấn luyện.

Trong thuật toán Mini Batch KMeans, mỗi lần lặp chỉ một lô dữ liệu được sử dụng để cập nhật các tâm cụm (cluster centroids). Điều này giúp giảm thiểu thời gian huấn luyện và bộ nhớ yêu cầu so với thuật toán KMeans truyền thống.

Mini Batch KMeans có thể được sử dụng để giải quyết các bài toán phân cụm với dữ liệu quá lớn để xử lý bằng thuật toán KMeans truyền thống. Tuy nhiên, do chỉ sử dụng một phần dữ liệu để cập nhật các tâm cụm, kết quả của Mini Batch KMeans có thể kém chính xác hơn so với KMeans truyền thống. [8]

– **Gradient Boosting:**

Một thuật toán học tăng cường (boosting) có thể được sử dụng để giải quyết các bài toán phân loại và hồi quy. Nó hoạt động bằng cách kết hợp nhiều mô hình yếu (weak learners) để tạo ra một mô hình mạnh (strong learner). [9]

– **Cat Boost:**

Một thuật toán gradient boosting trên cây quyết định (decision trees) được phát triển bởi Yandex. Nó có thể được sử dụng để giải quyết các bài toán phân loại và hồi quy và được thiết kế để xử lý dữ liệu danh mục (categorical data) hiệu quả. [10]

### 3. Cài đặt thực nghiệm

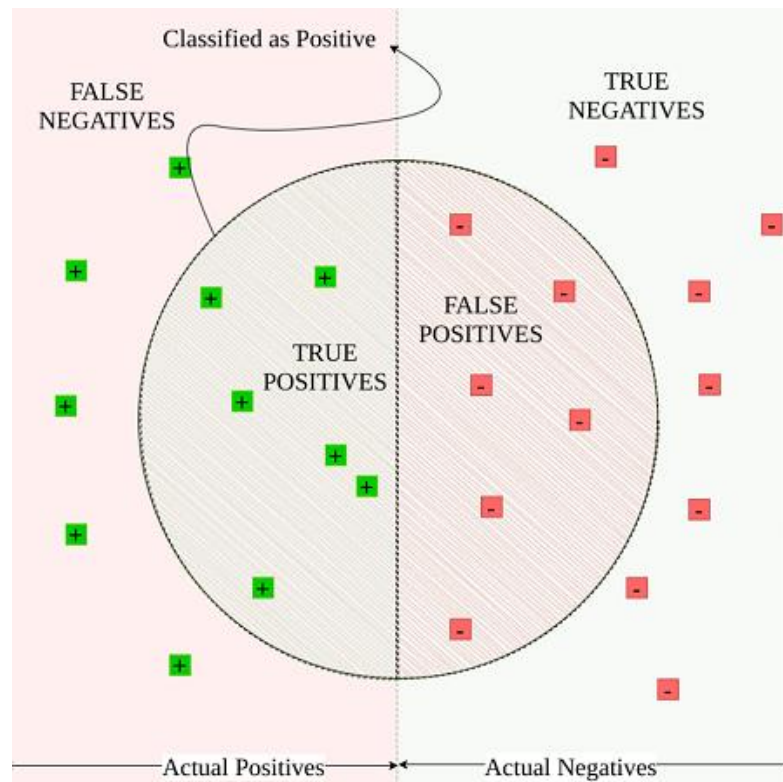
#### 3.1 Dataset

Dữ liệu được đưa vào sử dụng bao gồm 8044 dòng và 21 cột. Được chia làm thành tập train và tập test với tỉ lệ 0.8 và 0.2 (tương ứng 6426 và 1069 dòng dữ liệu). Trong quá trình thực nghiệm mô hình chia theo phương pháp cross-validation với hệ số  $k = 5$ .

#### 3.2 Phương pháp đánh giá

Khi thực hiện bài toán phân loại, có 4 trường hợp của dự đoán có thể xảy ra:

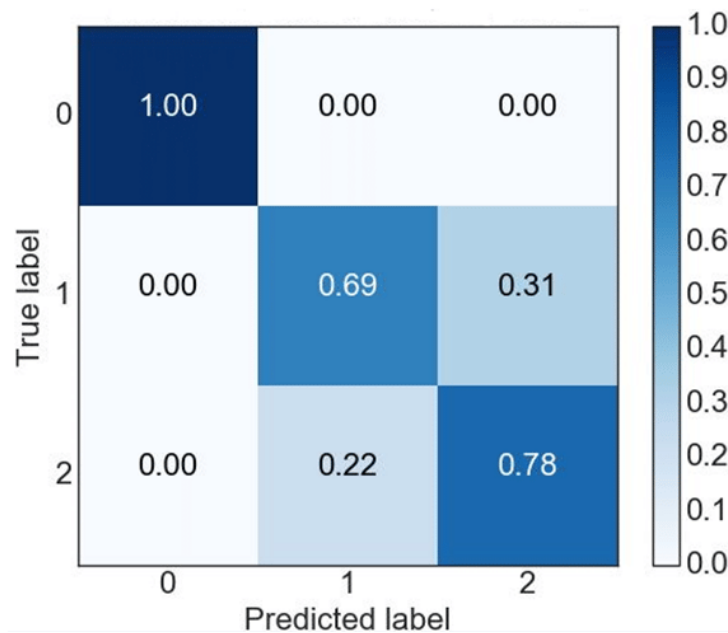
- True Positive (TP): đối tượng ở lớp Positive, mô hình phân đối tượng vào lớp Positive (dự đoán đúng).
- True Negative (TN): đối tượng ở lớp Negative, mô hình phân đối tượng vào lớp Negative (dự đoán đúng).
- False Positive (FP): đối tượng ở lớp Negative, mô hình phân đối tượng vào lớp Positive (dự đoán sai).
- False Negative (FN): đối tượng ở lớp Positive, mô hình phân đối tượng vào lớp Negative (dự đoán sai).



Hình 12 Minh họa các trường hợp dự đoán

### 3.2.1 Confusion matrix

Bốn trường hợp dự đoán trên thường được biểu diễn dưới dạng ma trận nhầm lẫn (confusion matrix). Confusion matrix là một công cụ quan trọng để đánh giá hiệu suất của các mô hình phân loại. Nó là một bảng hai chiều, trong đó các hàng tương ứng với các nhãn thực tế và các cột tương ứng với các nhãn được dự đoán bởi mô hình. Ma trận nhầm lẫn giúp đánh giá khả năng của mô hình phân loại các mẫu vào từng lớp.



Hình 13 Ví dụ confusion matrix – ma trận nhầm lẫn

Ma trận nhầm lẫn cung cấp một cái nhìn tổng quan về khả năng phân loại của mô hình, giúp việc điều chỉnh và cải thiện mô hình trở nên dễ dàng hơn. Đồng thời giúp các nhà phát triển mô hình hiểu được mô hình của họ đang phân loại mẫu trong từng lớp khác nhau như thế nào và loại trừ hoặc giảm thiểu một số sai sót phân loại.

### 3.2.2 Accuracy

Accuracy (độ chính xác) là độ đo đơn giản nhất để đánh giá một mô hình phân loại, được tính bằng là tỷ lệ giữa số điểm dữ liệu dự đoán đúng và tổng số điểm dữ liệu:

$$accuracy = \frac{\text{correct predictions}}{\text{total data points}} = \frac{TP + TN}{TP + TN + FP + FN}$$

Tỷ lệ accuracy càng cao, mô hình càng có hiệu suất tốt hơn trong việc dự đoán đúng các điểm dữ liệu mới. Tuy nhiên, độ đo accuracy có thể không phù hợp cho các bài toán mà số lượng các lớp khác nhau không cân bằng, một lớp có số lượng mẫu rất nhiều, trong khi các lớp khác có số lượng mẫu ít hơn. Trong trường hợp này, mô hình có thể dự đoán chính xác cho lớp có số lượng mẫu lớn hơn, nhưng lại không dự đoán chính xác cho các lớp khác.

### 3.2.3 F1\_score

F1-score là trung bình điều hòa (harmonic mean) của precision và recall (giả sử hai đại lượng này khác 0). F1-score được tính theo công thức:

$$F_1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Trong đó, Precision được định nghĩa là tỷ lệ số điểm Positive mô hình dự đoán đúng trên tổng số điểm mô hình dự đoán là Positive, Recall được định nghĩa là tỷ lệ số điểm Positive mô hình dự đoán đúng trên tổng số điểm thật sự là Positive (hay tổng số điểm được gán nhãn là Positive ban đầu).

## Precision

Of all **positive predictions**, how many are **really positive**?

$$\frac{TP}{TP + FP}$$

## Recall

Of all **real positive cases**, how many are **predicted positive**?

$$\frac{TP}{TP + FN}$$

		Real Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

		Real Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Hình 14 Precision và Recall

Precision càng cao, số điểm Positive mô hình dự đoán là Positive càng nhiều. Precision = 1, tất cả số điểm Positive mô hình dự đoán đúng là Positive, hay không có điểm Negative nào mô hình dự đoán nhầm là Positive. Recall càng cao, số điểm là Positive bị bỏ sót càng ít. Recall = 1, tức là tất cả số điểm Positive đều được mô hình nhận ra.

F1-score là độ đo kết hợp giữa precision và recall để đánh giá hiệu suất của mô hình phân loại. F1-score đạt giá trị cao khi precision và recall đều cao. Các giá trị F1-score thường nằm trong khoảng từ 0 đến 1, với giá trị càng cao cho thấy mô hình có hiệu suất phân loại càng tốt. F1-score cũng thường được sử dụng để so sánh hiệu suất của các mô hình phân loại khác nhau.

### 3.3 Phương pháp thực nghiệm

- Thuật toán Naive Bayes với hyperparameters: {'var\_smoothing': 1e-09}.
- Thuật toán SVM với hyperparameters: {'C': 1, 'gamma': 0.1, 'kernel': 'rbf'}.
- Thuật toán Gradient Boosting với hyperparameters: {'learning\_rate': 0.01, 'max\_depth': 5}.

- Thuật toán Cat Boost với hyperparameters: {iterations = 1500, learning\_rate = 0.02, depth = 6, l2\_leaf\_reg = 0.9, class\_weight = mảng % phân bố nhãn của các lớp }.
- Kết hợp giữa các mô hình thuật toán lại với nhau bao gồm (SVM, Logistic Regression, Random Forest, KNN, Decision Tree) phương pháp Voting Classifier để kết hợp nhiều mô hình với { KNN:  $n\_neighbors=5$ , Decision Tree:  $max\_depth=5$ , SVM:  $kernel='rbf'$ ,  $C=1.0$ , Logistic Regression:  $solver='liblinear'$ , RandomForest:  $n\_estimators=100$ ,  $max\_depth=10$ }.
- Thuật toán tăng cường: SGD với hyperparameters: {loss='log', learning\_rate = 'optimal', penalty='elasticnet', max\_iter=1000, tol=1e-3, random\_state=42}.
- Thuật toán tăng cường: MLP với hyperparameters: { hidden\_layer\_sizes=(200, 200)}.
- Thuật toán tăng cường: : PassiveAggressive.
- Thuật toán tăng cường: MiniBatchKMeans với hyperparameters: {n\_clusters=5, batch\_size=300}.
- Thuật Toán Random Forest kết hợp SMOTE: Hyperparameters: (n\_estimators=100, random\_state=42)

### 3.4 Kết quả thực nghiệm

	Accuracy	F1_score(macro)
Decision Tree	92.7433	72.01
<b>Random Forest</b>	95.0442	<b>76.02</b>
<b>Logistic Regression</b>	<b>95.3982</b>	75.23
KNN	94.3362	65.333
SVM	95.2212	66.322
Random Forest + SMOTE	83.8053	57.5453

*Bảng 1: Kết quả chạy thực nghiệm cho dự đoán điểm môn IT002 dựa trên các môn học trước.*

#### Nhận xét:

- Điểm các môn trước có thể dự đoán được kết quả môn học IT002 vì do đều là các môn cơ sở ngành và đều cùng chung 1 năm học không có độ thay đổi quá nhiều đối với sinh viên.

- Kết quả của mô hình dự đoán Logistics Regression có accuracy cao nhất 95.4982% vì dự đoán được nhiều điểm dữ liệu đúng nhiều nhất, bên cạnh đó Random Forest lại cho kết quả F1 lại cao nhất 76%.
- Ngoài ra kết quả accuracy cao đều trên 90% nhưng f1 lại dưới dao động từ 65 – 76% vì bị mất cân bằng nhãn giữa sinh viên bị rớt nhãn 0 và sinh viên qua môn nhãn 1 (nhãn 1 có số lượng gấp 10 lần nhãn 0) nên khi cân bằng nhãn bằng Smote và chạy mô hình thì kết quả đã thấp xuống 83.8053% đối với accuracy và 57.5453 với f1.

	Học kỳ 1	Học kỳ 2	Học kỳ 3	Học kỳ 4	Học kỳ 5	Học kỳ 6	Học kỳ 7	Học kỳ 8
SVM	0.430	<b>0.528</b>	0.555	0.619	0.590	0.636	0.552	<b>0.699</b>
Logistic Regression	0.427	0.436	0.496	0.590	0.535	0.598	0.479	0.666
Random Forest	0.406	0.484	0.519	0.592	0.550	0.602	0.551	0.678
KNN	0.335	0.405	0.457	0.543	0.499	0.561	0.497	0.601
Decision Tree	0.407	0.482	0.5	0.548	0.518	0.542	0.479	0.609
<b>Ensemble</b>	<b>0.431</b>	0.505	<b>0.563</b>	<b>0.625</b>	0.602	<b>0.637</b>	0.568	0.698
Naive bayes	0.41	0.475	0.508	0.513	0.547	0.589	0.502	0.623
Gradient Boosting	0.4	0.521	<b>0.563</b>	0.567	<b>0.609</b>	0.633	<b>0.578</b>	0.67
SGD	0.4	0.33	0.38	0.36	0.23	0.26	0.23	0.26
MLP	0.39	0.37	0.45	0.47	0.45	0.42	0.36	0.47
PassiveAggressive	0.39	0.19	0.32	0.16	0.19	0.25	0.34	0.21
MiniBatchKMeans	0.17	0.19	0.18	0.15	0.14	0.14	0.16	0.16
Cat Boost	0.393	0.463	0.531	0.617	0.594	0.597	0.569	0.687

*Bảng 2: Kết quả chạy thực nghiệm cho dựa vào tiền đề và xếp loại của các học kì trước dự toán học kì tiếp theo (từ hk1 – hk 8) được đo theo accuracy.*

### **Nhận xét:**

- Các yếu tố về nền tảng giáo dục trước đó có thể không thực sự ảnh hưởng đến kết quả của điểm trung bình học kì của số đông sinh viên và phần đông sinh viên không có tính ổn định giữa các kì nên dẫn đến khó dự đoán điểm trung bình học kì dựa trên tiền đề và điểm trung bình của các kì trước, nhưng các kì sau đa số



sinh viên đã có thể ổn định hơn ở điểm trung bình, các điểm kì trung bình trước đó có thể là các tác động với kết quả dự đoán kì sau.

- Có 3 mô hình cho kết quả tốt nhất là GradientBoosting, Ensemble, SVM bên cạnh đó mô hình có tính ổn định và cao là Ensemble.
- Có các mô hình học tăng cường như SGD, MLP, PassiveAggressive, MiniBatchKMeans cho kết quả rất thấp và không có tính ổn định nhưng thời gian chạy rất nhanh chỉ vài giây cho mỗi lần chạy.
- Ngoài ra vì sự phân bố các nhãn không đồng đều giữa 5 nhãn và các kì 7, 8 phần đông sinh viên chỉ đăng kí ít hoặc đã làm khóa luận nên điểm trung bình có sự khác biệt lớn ở các nhãn.

#### 4. Kết luận và hướng phát triển

Bài toán các yếu tố về nền tảng giáo dục trước đó có thể không thực sự ảnh hưởng đến kết quả của điểm trung bình học kì kết quả đạt được từng kì là khác nhau và dao động từ 43% - 69.9% với các mô hình SVM, Ensemble, Gradient Boosting. Ngoài ra đối với điểm các môn trước (môn cơ sở ngành khác) có thể dự đoán được kết quả môn học IT002 đạt kết quả phân loại khá cao với kết quả mô hình Logistic Regression 95.3982% đối với độ đo accuracy và mô hình Random Forest 76.02% với độ đo f1-score. Đối với sinh viên, kết quả dự đoán này có thể giúp họ cải thiện khả năng đạt được điểm số tốt trong IT002 bằng cách học tốt trong các môn học cơ sở ngành trước đó và tham gia tích cực vào các hoạt động lớp học. Đối với giảng viên, việc dự đoán này cung cấp một công cụ hữu ích để xác định những sinh viên có thể cần sự hỗ trợ hoặc can thiệp thêm, giúp nâng cao chất lượng giảng dạy. Góp phần vào lĩnh vực khai thác dữ liệu giáo dục bằng cách phát triển một phương pháp mới để dự đoán điểm số của sinh viên dựa trên các khóa học trước và hiệu suất học tập. Phương pháp này có thể giúp cả sinh viên và giảng viên cải thiện kết quả và trải nghiệm học tập của họ.

Hướng phát triển:

- Mở rộng bộ dữ liệu để bao gồm nhiều trường đại học và khóa học.
- Bổ sung thêm các biến vào mô hình dự đoán, chẳng hạn như các đặc tính cá nhân hoặc điểm phản hồi.
- Đánh giá hiệu suất của mô hình trong các bối cảnh và kịch bản khác nhau.

- Thêm các yếu tố tâm lý và tính cách như bài test MBTI để đánh giá một cách khách quan hơn.
- viên có thể đưa ra những lựa chọn phù hợp dựa trên đánh giá khách quan về điểm số và những thống kê từ các khóa học trước đó.

## 5. Tài liệu tham khảo

- [1] M. H. R. J. L. a. A. J. Sweeney, "Next-term student performance prediction: A recommender systems approach," arXiv:1604.01840, 2016.
- [2] A. T. L. W. N. T. T. A. G. a. E.-P. L. Widjaja, "Next-term grade prediction: A machine learning approach," (2020): 700..
- [3] A. M. S. a. A. A.-E. Nabil, "Prediction of students' academic performance based on courses' grades using deep neural networks," 140731-140746, IEEE Access 9 (2021).
- [4] "SMOTE for Imbalanced Classification with Python",  
"https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/?fbclid=IwAR0sVHNWFU9bAtvxkRu70qDaR89xhwDHkBA05VwBsiQ0PwoVJyBO-MZ6L6I," ( Truy cập lần cuối: 28/05/2023).
- [5] S. & J. N. Musa Baig, "Stochastic Gradient Descent Algorithm (SGD).," 2022.
- [6] Z. D. D. R. S. a. Q. V. L. Hanxiao Liu, "Pay Attention to MLPs," arXiv:2105.08050, 2021.
- [7] T. S. a. J. Zhu, "Online Bayesian Passive-Aggressive Learning },"  
https://doi.org/10.3389/fpsyg.2021.579183, 2013.
- [8] B. A. Javier, "K-means vs Mini Batch K-means: a comparison," arXiv:1602.02934, 2013.
- [9] D. L. T. L. M. W. Zhiyuan He, "Gradient Boosting Machine: A Survey," arXiv:1908.06951, 2019.
- [10] G. G. A. V. A. V. D. A. G. Liudmila Prokhorenkova, "CatBoost: unbiased boosting with  
] categorical features," https://doi.org/10.48550/arXiv.1706.09516, 2019.
- [11] "Scikitlearn," [Online]. Available: https://scikit-learn.org/stable/.  
]
- [12] "Coursera," [Online]. Available: https://www.coursera.org/learn/machine-learning.  
]

**6. Phân công công việc**

Nội dung	Phân công công việc	Thành viên	Đánh giá
<ul style="list-style-type: none"> <li>Tìm hiểu dữ liệu</li> <li>Làm sạch dữ liệu</li> <li>Phân tích dữ liệu</li> <li>Mô tả dữ liệu</li> </ul>	01.sinhvien & 02.diem & Sinhvien_dtb_hocky	Trúc	Hoàn thành
	03.sinhvien_chungchi & 04.xeploaiav & diem_Thu	Hùng	Hoàn thành
	05.ThiSinh & 06.giayxacnhan & Sinhvien_dtb_toankhoa	Hà	Hoàn thành
	08.XLHV & 10.diemrl & diemrl	Hoàng	Hoàn thành
	12.baoluu & 14.totnghiep & uit_hocphi_miengiam	Duyên	Hoàn thành
Bài tập thực hành 1	Phân tích vấn đề và dữ liệu theo hướng đề tài thực hiện	Cả nhóm	Hoàn thành
	Tổng hợp file mô tả dữ liệu	Trúc	Hoàn thành
	Phân tích và xử lý dữ liệu theo bài toán đã chọn	Duyên, Hà, Hùng	Hoàn thành
	Vẽ biểu đồ phân tích các thuộc tính input, output	Hà, Hoàng	Hoàn thành
	Làm Powerpoint bài tập thực hành 1	Duyên, Hà	Hoàn thành
Viết thuyết minh đề tài	P1. Mô tả đề tài	Trúc	Hoàn thành
	P2. Tổng quan	Trúc, Hoàng	Hoàn thành
	P3. Mục tiêu	Hùng	Hoàn thành
	P4. Nội dung và phương pháp thực hiện	Duyên	Hoàn thành
	P5. Kết quả dự kiến	Hà	Hoàn thành
	P7. Bảng phân công công việc	Duyên	Hoàn thành
Bài tập thực hành 2	Chuyển cột diemthi thành biến phân loại, thêm các cột kết quả	Duyên	Hoàn thành

	học tập của các kì tiếp theo vào dữ liệu		
	Chạy thực nghiệm ít nhất 2 mô hình trên bộ dữ liệu	Hà	Hoàn thành
	Viết mô tả dữ liệu	Hùng	Hoàn thành
	Tổng hợp file + nộp bài	Duyên	Hoàn thành
	Làm powerpoint dựa theo thuyết minh đề tài	Trúc	Hoàn thành
Báo cáo cuối kì	Điền khuyết dữ liệu.		
	Viết báo cáo phần phương pháp tiền xử lí.	Duyên	Hoàn thành
	Viết báo cáo phần mô tả bộ dữ liệu	Hùng	Hoàn thành
	Viết báo cáo phần lí thuyết các độ đo được sử dụng	Trúc	Hoàn thành
	Chạy thực nghiệm bộ dữ liệu trên các thuật toán	Hoàng Hà Hùng	Hoàn thành
	Viết báo cáo phần phương pháp đề xuất và phân tích kết quả		
	Viết báo cáo phần kết luận và hướng phát triển	Trúc	Hoàn thành
Thuyết trình đề tài	Làm slide	Cả nhóm	Hoàn thành
	Thuyết trình + Quay video	Cả nhóm	Hoàn thành
	Edit video	Hoàng	Hoàn thành