Data Mining Challenge

IE343 Statistical Machine Learning

OBJECTIVE

- Let’s apply the knowledge we learned during the course to an actual data mining problem!

- Goal: to **predict students' academic performance a.k.a grades** based on their characteristics
  - We will provide you with the characteristics and grades of 80% (training data) of the students
  - Your task will be to build a machine learning (ML) model that can predict the grades of the remaining 20% (test data)
  - In essence, a classification task with 7 classes, each corresponding to a different grading

- You are free to try **any ML method**, not limited to those we learned in our lectures

- **Analyzing & engineering the given data** will further improve your ML methods’ performance. Feel free to adapt any kind of feature engineering skill during the competition!

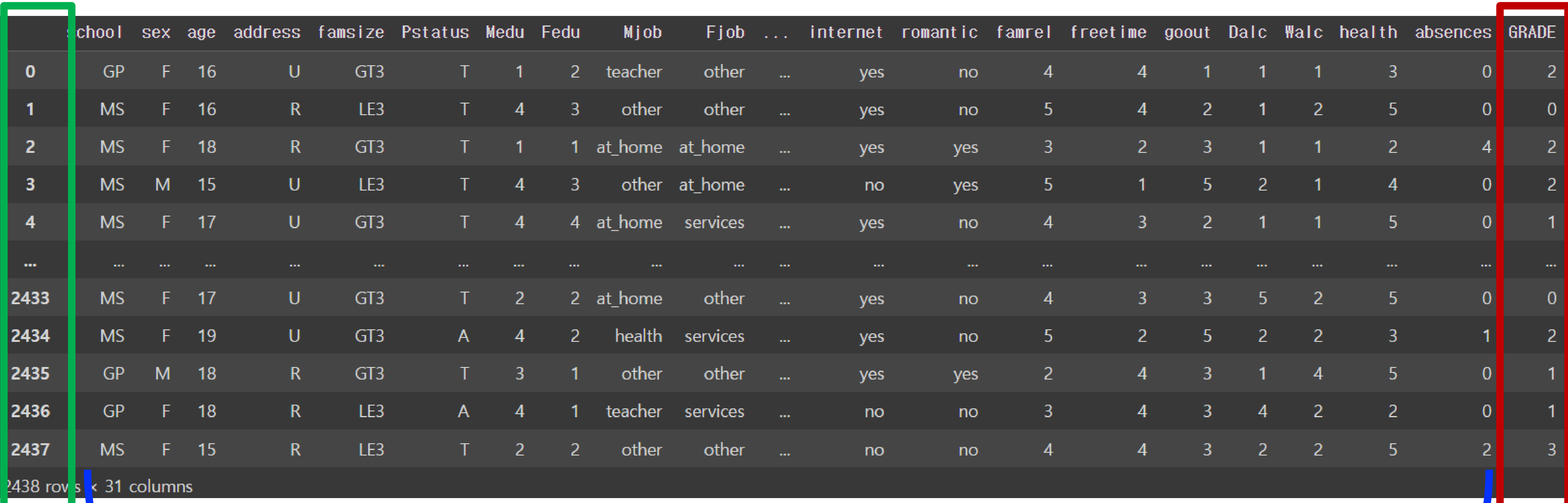
OBJECTIVE

- Challenge timeline: 4/22 00:00 ~ 5/27 23:59
  - Competition link: https://www.kaggle.com/t/e33dacfa8b1340339fe45fdc29f882f0

DATA DESCRIPTION

- **train_data.csv** - the training data (80% of full data = 2438 students)

Target column



	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	...	internet	romantic	famrel	freetime	goout	Dalc	Walc	health	absences	GRADE
0	GP	F	16	U	GT3	T	1	2	teacher	other	...	yes	no	4	4	1	1	1	3	0	2
1	MS	F	16	R	LE3	T	4	3	other	other	...	yes	no	5	4	2	1	2	5	0	0
2	MS	F	18	R	GT3	T	1	1	at_home	at_home	...	yes	yes	3	2	3	1	1	2	4	2
3	MS	M	15	U	LE3	T	4	3	other	at_home	...	no	yes	5	1	5	2	1	4	0	2
4	MS	F	17	U	GT3	T	4	4	at_home	services	...	yes	no	4	3	2	1	1	5	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2433	MS	F	17	U	GT3	T	2	2	at_home	other	...	yes	no	4	3	3	5	2	5	0	0
2434	MS	F	19	U	GT3	A	4	2	health	services	...	yes	no	5	2	5	2	2	3	1	2
2435	GP	M	18	R	GT3	T	3	1	other	other	...	yes	yes	2	4	3	1	4	5	0	1
2436	GP	F	18	R	LE3	A	4	1	teacher	services	...	no	no	3	4	3	4	2	2	0	1
2437	MS	F	15	R	LE3	T	2	2	other	other	...	no	no	4	4	3	2	2	5	2	3

2438 rows × 31 columns

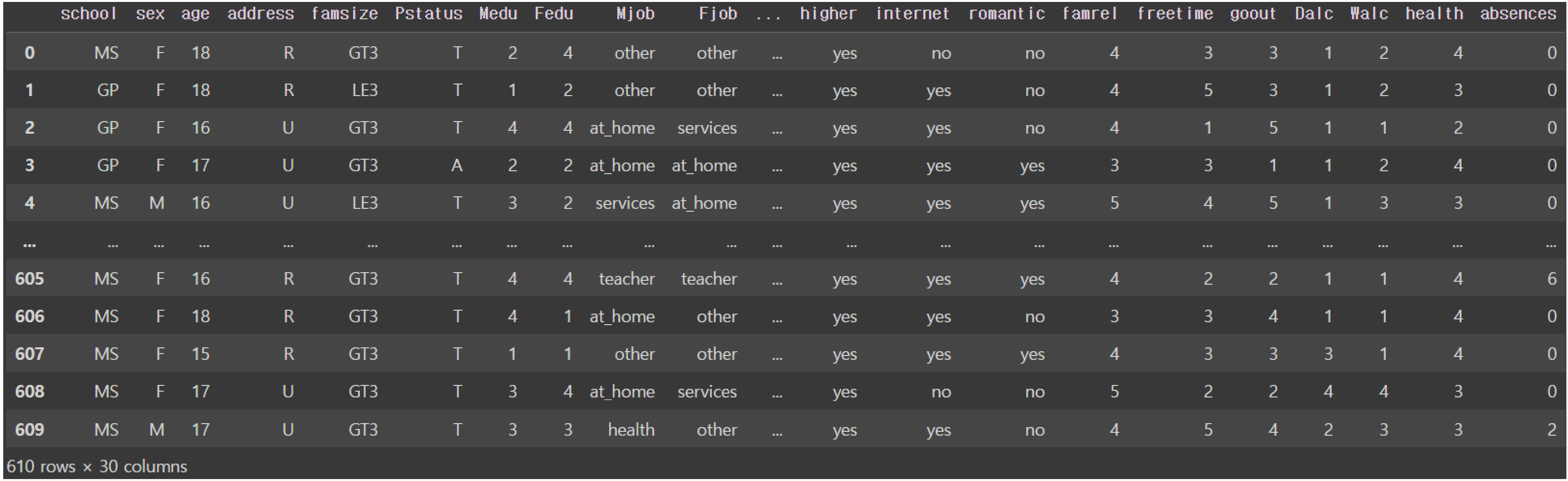
Each row represents an individual student

Characteristics of each student

DATA DESCRIPTION

- **test_data.csv** - the test data (20% of full data = 610 students)

No target column



	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	...	higher	internet	romantic	famrel	freetime	goout	Dalc	Walc	health	absences
0	MS	F	18	R	GT3	T	2	4	other	other	...	yes	no	no	4	3	3	1	2	4	0
1	GP	F	18	R	LE3	T	1	2	other	other	...	yes	yes	no	4	5	3	1	2	3	0
2	GP	F	16	U	GT3	T	4	4	at_home	services	...	yes	yes	no	4	1	5	1	1	2	0
3	GP	F	17	U	GT3	A	2	2	at_home	at_home	...	yes	yes	yes	3	3	1	1	2	4	0
4	MS	M	16	U	LE3	T	3	2	services	at_home	...	yes	yes	yes	5	4	5	1	3	3	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
605	MS	F	16	R	GT3	T	4	4	teacher	teacher	...	yes	yes	yes	4	2	2	1	1	4	6
606	MS	F	18	R	GT3	T	4	1	at_home	other	...	yes	yes	no	3	3	4	1	1	4	0
607	MS	F	15	R	GT3	T	1	1	other	other	...	yes	yes	yes	4	3	3	3	1	4	0
608	MS	F	17	U	GT3	T	3	4	at_home	services	...	yes	no	no	5	2	2	4	4	3	0
609	MS	M	17	U	GT3	T	3	3	health	other	...	yes	yes	no	4	5	4	2	3	3	2

610 rows × 30 columns

DATA DESCRIPTION

- **‘GRADE’** column **(this is the value you need to predict)**
  - Student's grade (0 – Fail, 1 – Poor, 2 – Bad, 3 – Average, 4 – Good, 5 – Excellent, 6 – Outstanding)

- **‘age’** column: student’s age

- **‘traveltime’** column: travel time from home to school

- **‘absences’** column: number of school absences

- Further details for all columns can be found in the ‘Data’ section on the competition page

EVALUATION

- Your model will be evaluated on **Classification Accuracy**, i.e.,

$$\frac{\#\ \text{of students whose grades were correctly predicted}}{\text{total}\ \#\ \text{of students whose grades were predicted}}$$

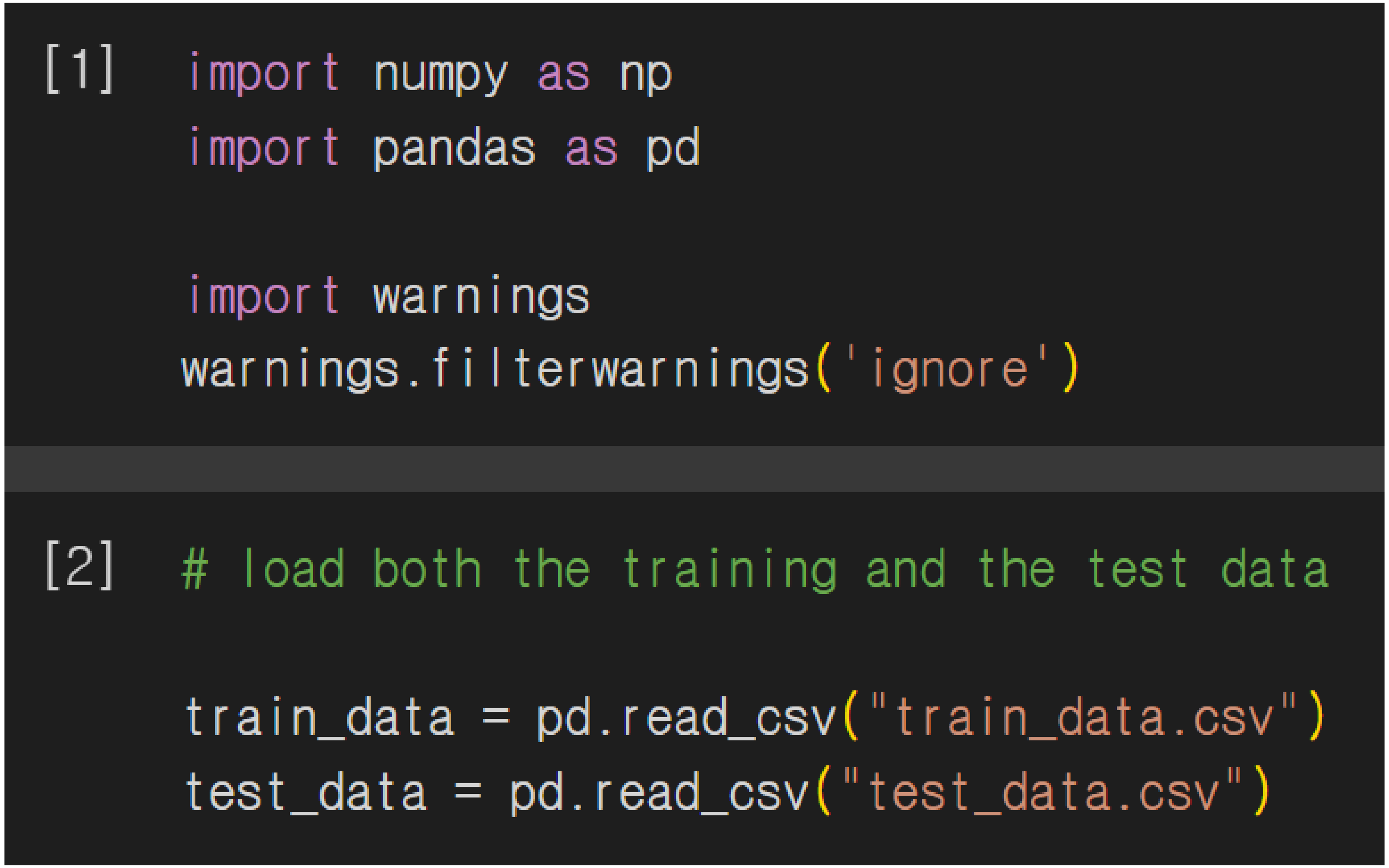
TUTORIAL

- In a nutshell
  - Train your model on the training data
  - Predict between 7 different classes on the test data
  - Then submit your predictions on Kaggle

- Refer to sample_code.ipynb

TUTORIAL

- Import necessary libraries, then load train_data.csv & test_data.csv into your workspace

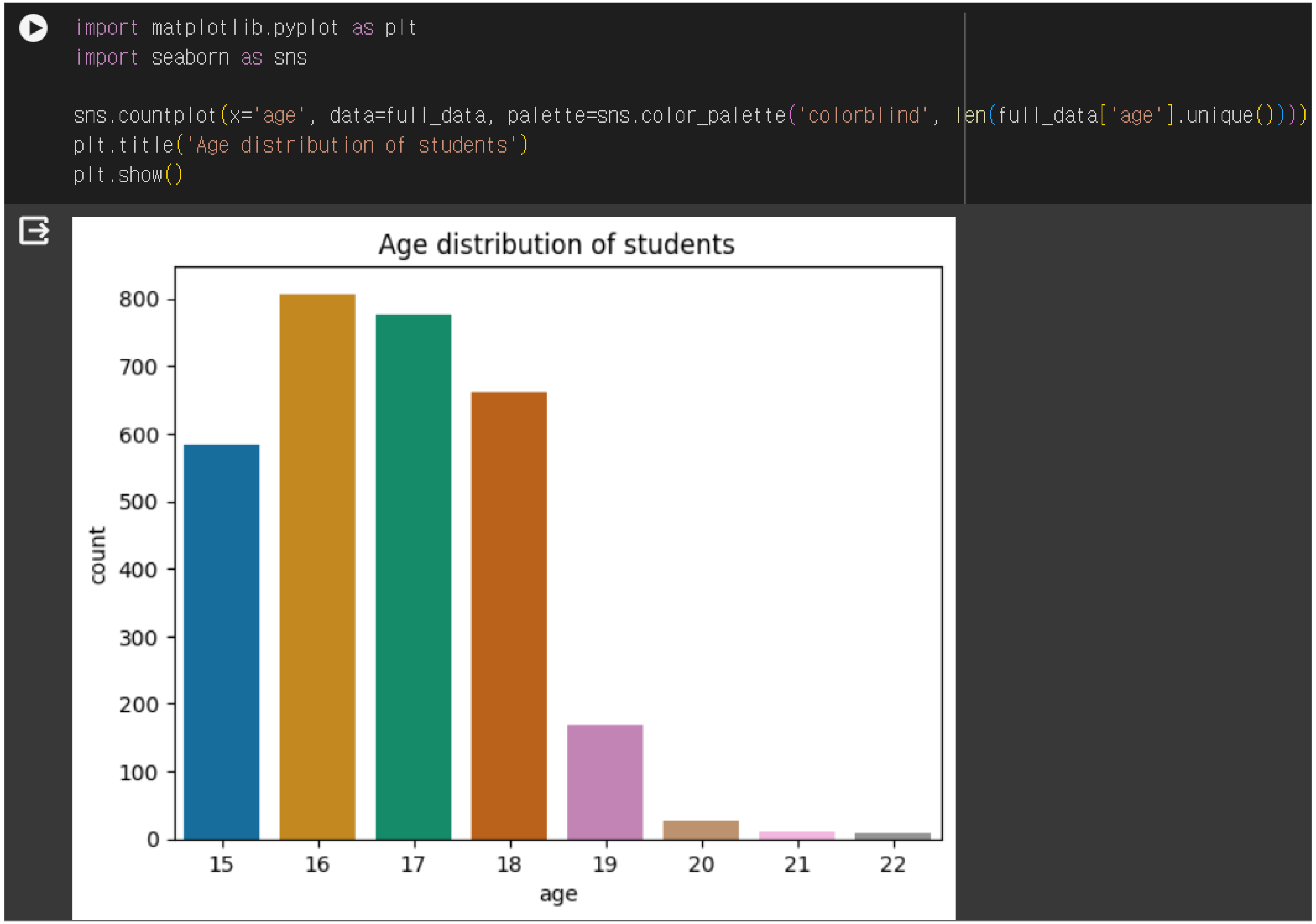


```
[1]  import numpy as np
     import pandas as pd

     import warnings
     warnings.filterwarnings('ignore')
```

```
[2]  # load both the training and the test data

     train_data = pd.read_csv("train_data.csv")
     test_data = pd.read_csv("test_data.csv")
```

TUTORIAL

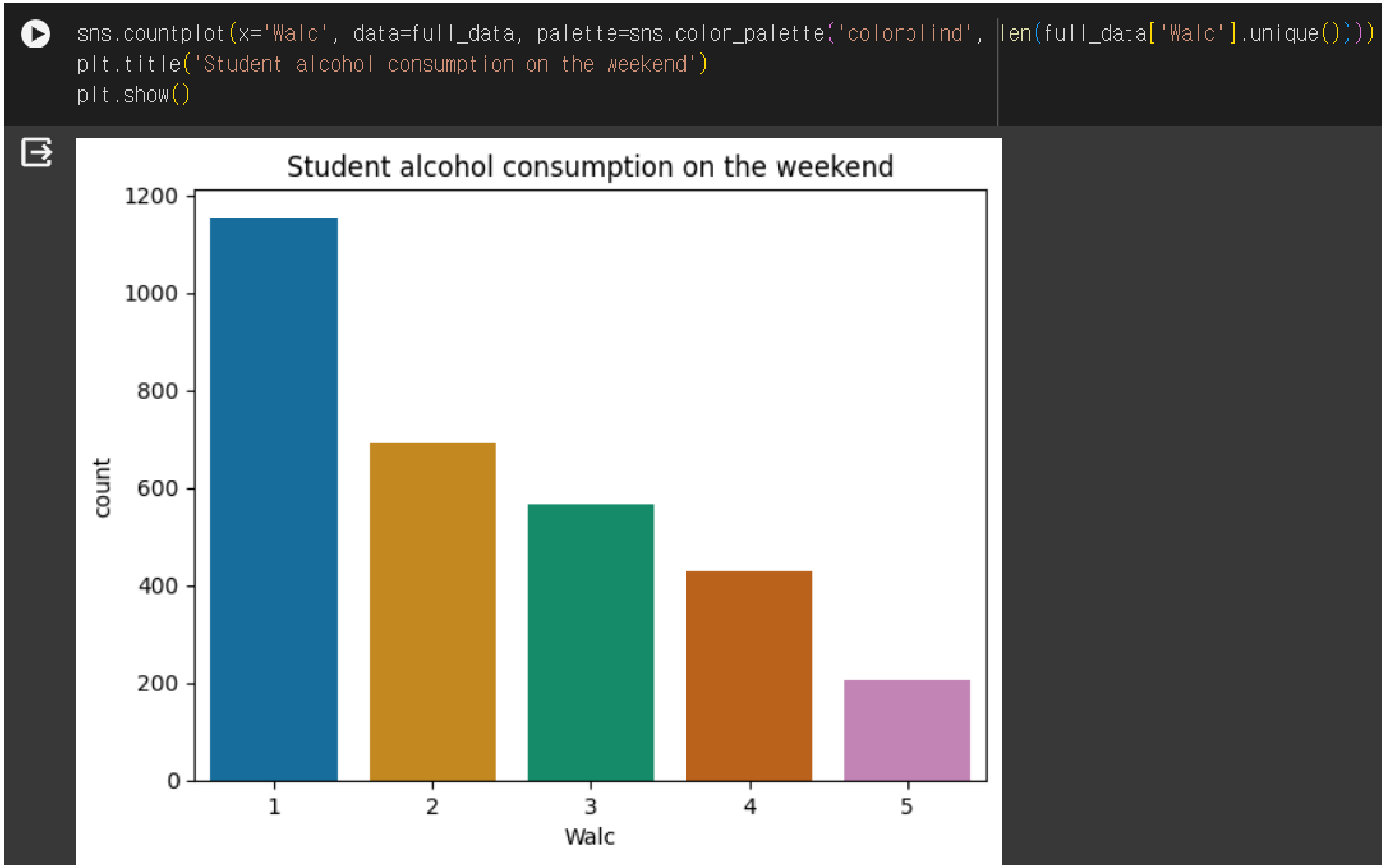
- Conduct extensive feature analysis



```
import matplotlib.pyplot as plt
import seaborn as sns

sns.countplot(x='age', data=full_data, palette=sns.color_palette('colorblind', len(full_data['age'].unique())))
plt.title('Age distribution of students')
plt.show()
```

TUTORIAL

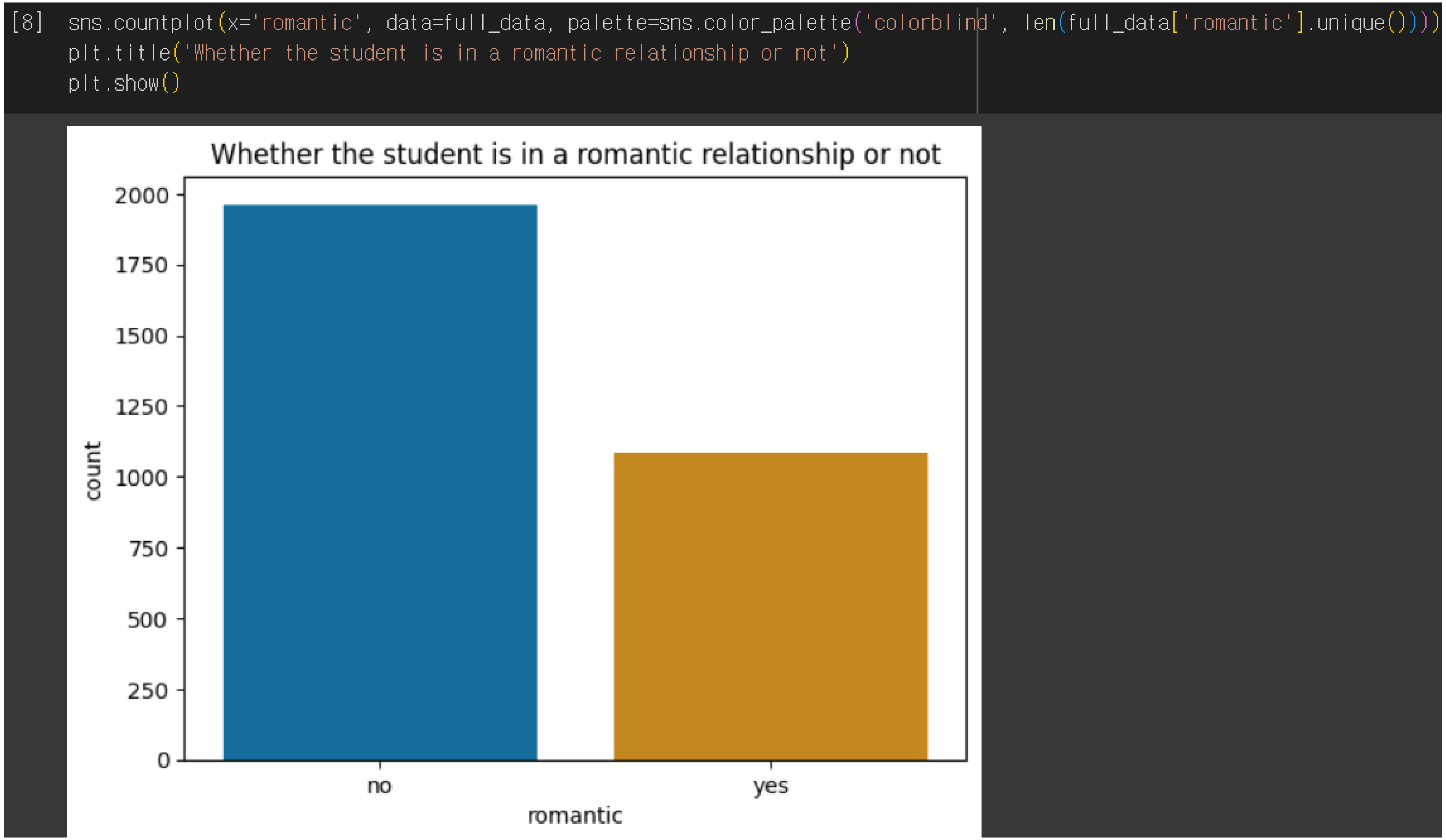
- Conduct extensive feature analysis



```
sns.countplot(x='Walc', data=full_data, palette=sns.color_palette('colorblind', len(full_data['Walc'].unique())))
plt.title('Student alcohol consumption on the weekend')
plt.show()
```

TUTORIAL

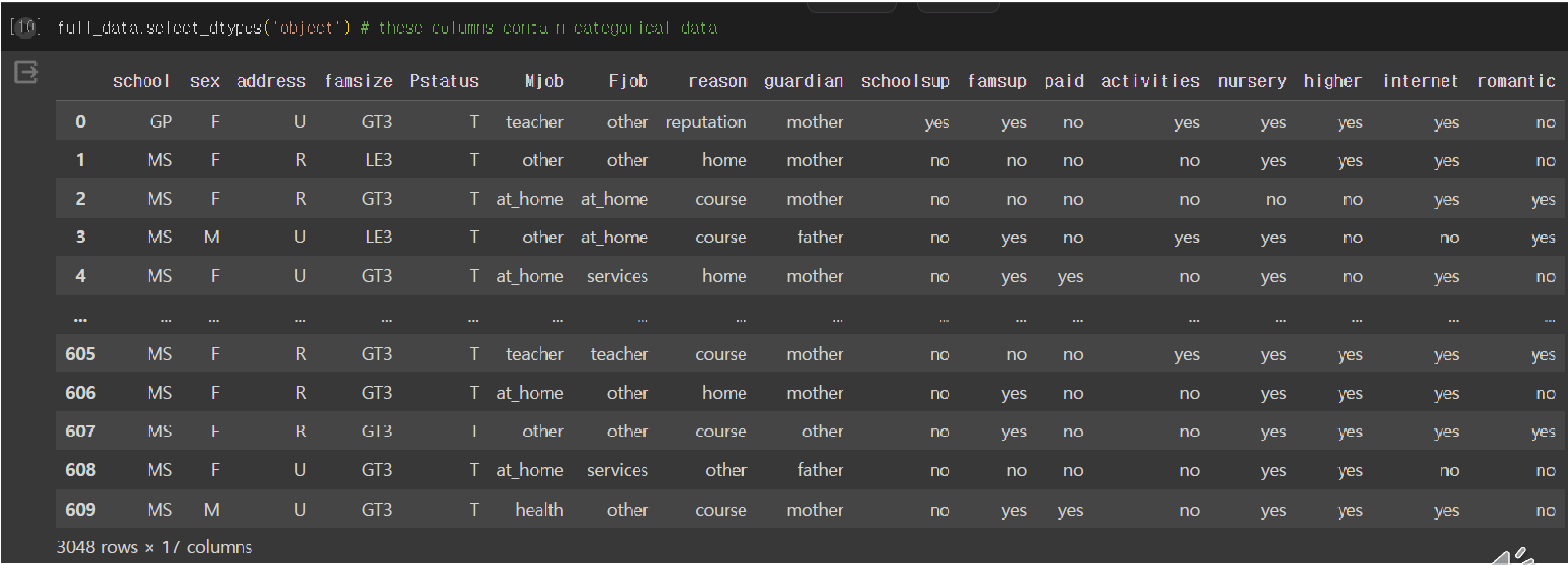
- Conduct extensive feature analysis



```
[8]  sns.countplot(x='romantic', data=full_data, palette=sns.color_palette('colorblind', len(full_data['romantic'].unique())))
     plt.title('Whether the student is in a romantic relationship or not')
     plt.show()
```

TUTORIAL

- Conduct extensive feature analysis



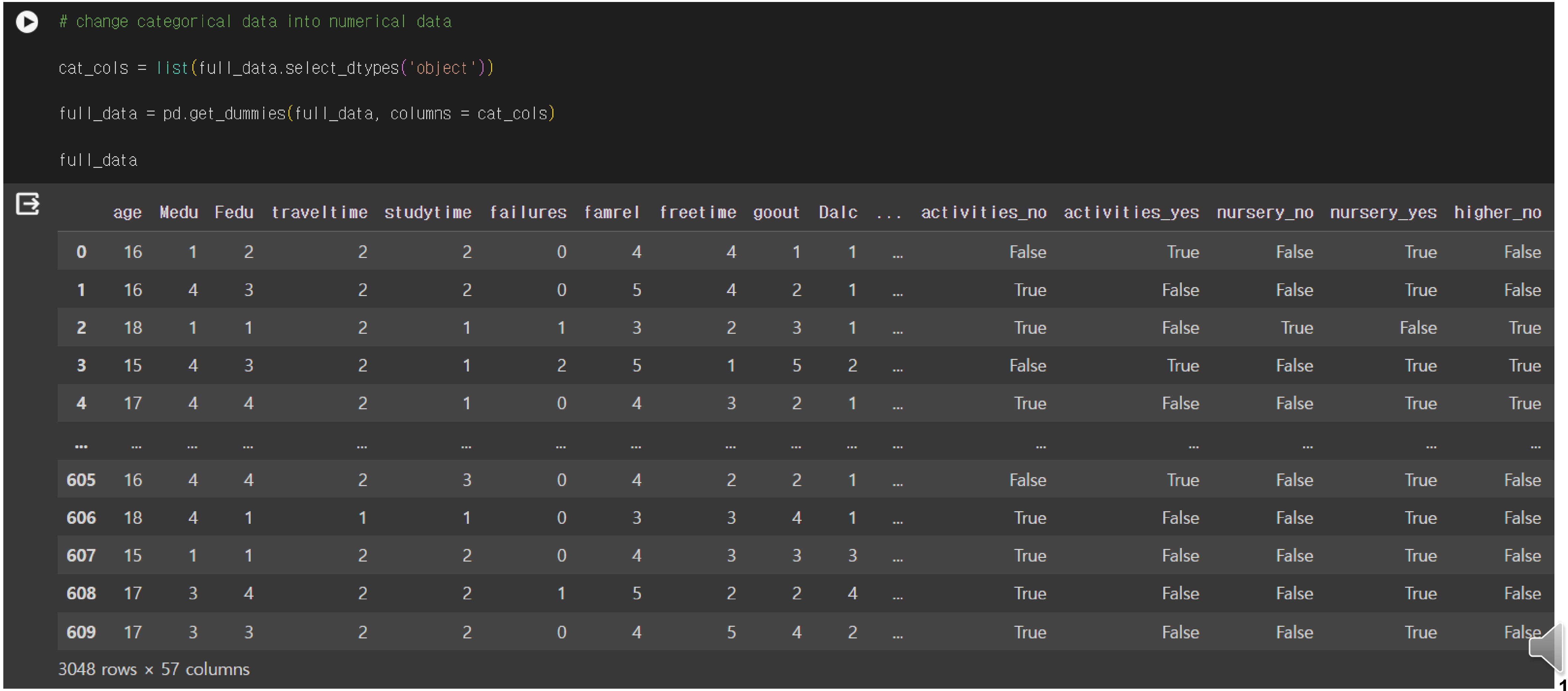
```
[10] full_data.select_dtypes('object') # these columns contain categorical data
```

	school	sex	address	famsize	Pstatus	Mjob	Fjob	reason	guardian	schoolsup	famsup	paid	activities	nursery	higher	internet	romantic
0	GP	F	U	GT3	T	teacher	other	reputation	mother	yes	yes	no	yes	yes	yes	yes	no
1	MS	F	R	LE3	T	other	other	home	mother	no	no	no	no	yes	yes	yes	no
2	MS	F	R	GT3	T	at_home	at_home	course	mother	no	no	no	no	no	no	yes	yes
3	MS	M	U	LE3	T	other	at_home	course	father	no	yes	no	yes	yes	no	no	yes
4	MS	F	U	GT3	T	at_home	services	home	mother	no	yes	yes	no	yes	no	yes	no
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
605	MS	F	R	GT3	T	teacher	teacher	course	mother	no	no	no	yes	yes	yes	yes	yes
606	MS	F	R	GT3	T	at_home	other	home	mother	no	yes	no	no	yes	yes	yes	no
607	MS	F	R	GT3	T	other	other	course	other	no	yes	no	no	yes	yes	yes	yes
608	MS	F	U	GT3	T	at_home	services	other	father	no	no	no	no	yes	yes	no	no
609	MS	M	U	GT3	T	health	other	course	mother	no	yes	yes	no	yes	yes	yes	no

3048 rows × 17 columns

TUTORIAL

- Conduct extensive feature analysis



```python
# change categorical data into numerical data

cat_cols = list(full_data.select_dtypes('object'))

full_data = pd.get_dummies(full_data, columns = cat_cols)

full_data
```

```
     age  Medu  Fedu  traveltime  studytime  failures  famrel  freetime  goout  Dalc  ...  activities_no  activities_yes  nursery_no  nursery_yes  higher_no
0     16     1     2           2          2         0       4         4      1     1  ...          False            True       False         True      False
1     16     4     3           2          2         0       5         4      2     1  ...           True           False       False         True      False
2     18     1     1           2          1         1       3         2      3     1  ...           True           False        True        False       True
3     15     4     3           2          1         2       5         1      5     2  ...          False            True       False         True       True
4     17     4     4           2          1         0       4         3      2     1  ...           True           False       False         True       True
...  ...   ...   ...         ...        ...       ...     ...       ...    ...   ...  ...            ...             ...         ...          ...        ...
605   16     4     4           2          3         0       4         2      2     1  ...          False            True       False         True      False
606   18     4     1           1          1         0       3         3      4     1  ...           True           False       False         True      False
607   15     1     1           2          2         0       4         3      3     3  ...           True           False       False         True      False
608   17     3     4           2          2         1       5         2      2     4  ...           True           False       False         True      False
609   17     3     3           2          2         0       4         5      4     2  ...           True           False       False         True      False
3048 rows × 57 columns
```

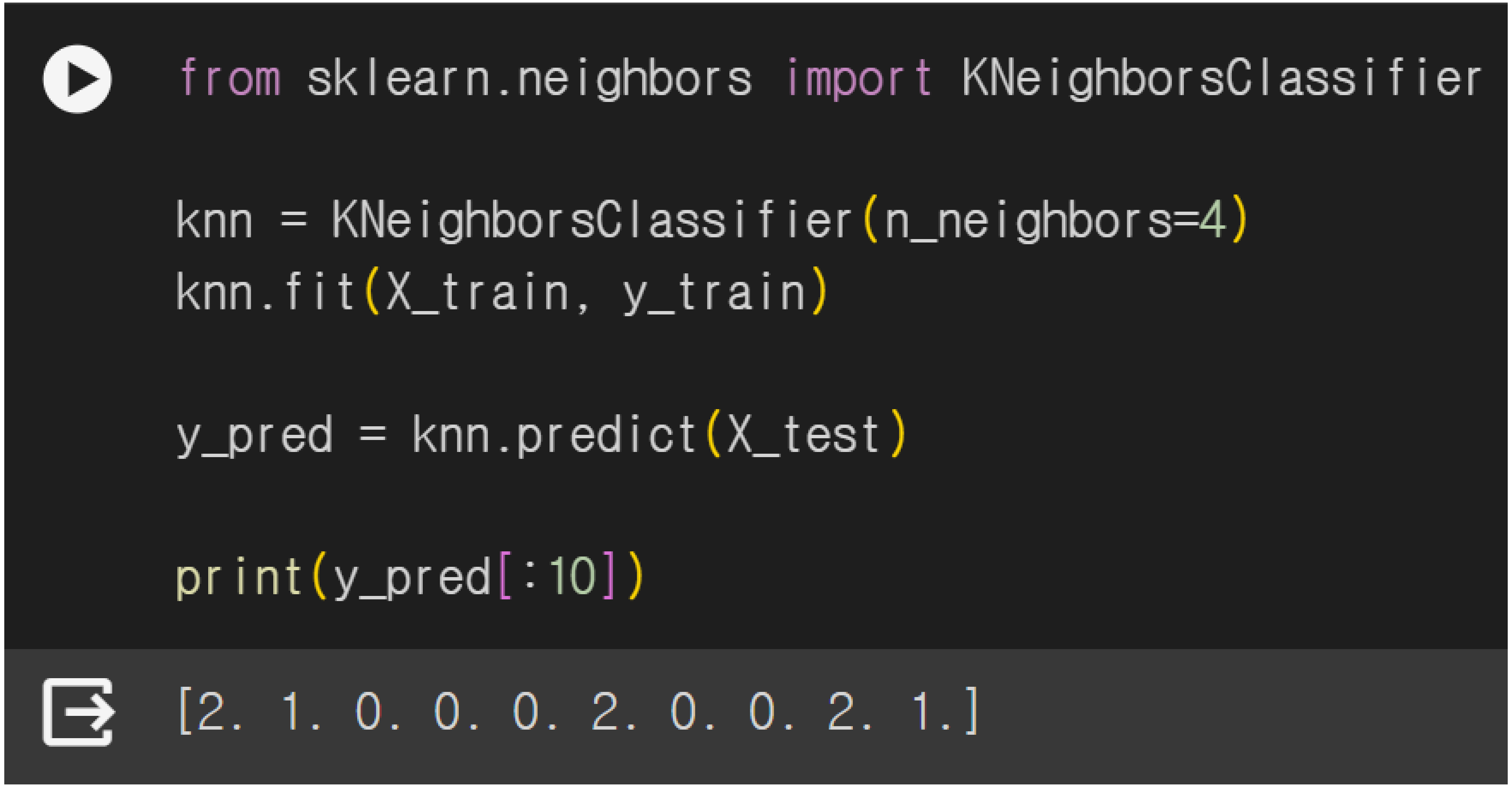
TUTORIAL

- Conduct extensive feature analysis
  - Data types
  - Skewness in distribution / class imbalance
  - Correlation between features
  - Difference in range between features
  - Etc.

- **Engineer** the given data!
  - Transform categorical data into numerical data
  - Log-transform values / add or delete rows
  - Feature selection
  - Scaling
  - Etc.

TUTORIAL

- Train a model and predict on the test data (should return 610 predictions)



```python
from sklearn.neighbors import KNeighborsClassifier

knn = KNeighborsClassifier(n_neighbors=4)
knn.fit(X_train, y_train)

y_pred = knn.predict(X_test)

print(y_pred[:10])
```

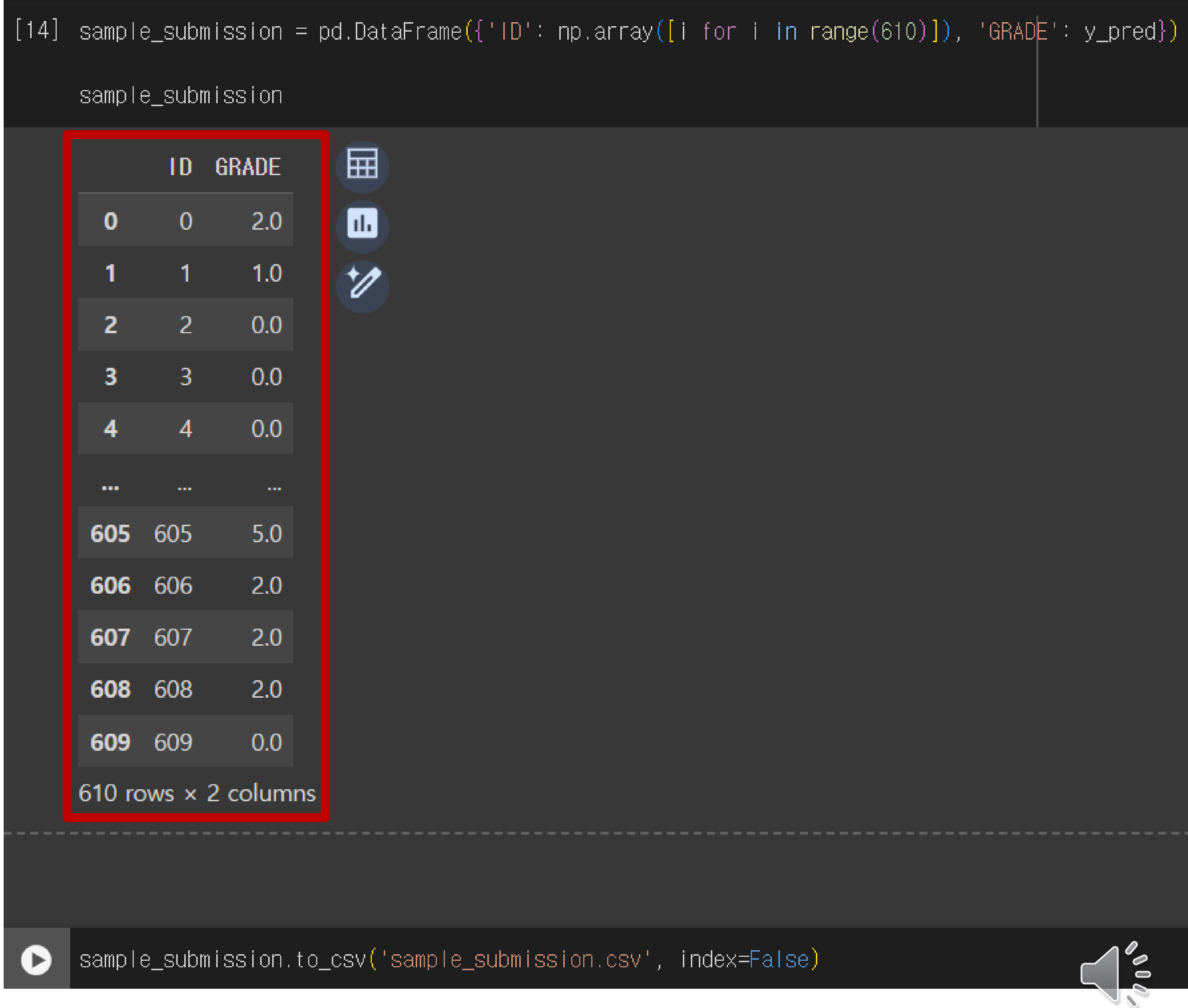
```
[2. 1. 0. 0. 0. 2. 0. 0. 2. 1.]
```

TUTORIAL

- Save your predictions into a CSV file

- Your submission file **MUST** look like this!
  - Two columns – ‘ID’ & ‘GRADE’
  - 610 rows
  - ‘ID’: index of the test students, ranging from 0 ~ 609 (fixed)
  - ‘GRADE’: your predictions per student, can change depending on your model
  - Check shape, values and header
  - File name is up to you

- Otherwise, errors will occur when you submit



```
[14] sample_submission = pd.DataFrame({'ID': np.array([i for i in range(610)]), 'GRADE': y_pred})
     sample_submission
```

```
       ID  GRADE
  0     0    2.0
  1     1    1.0
  2     2    0.0
  3     3    0.0
  4     4    0.0
 ...  ...    ...
605   605    5.0
606   606    2.0
607   607    2.0
608   608    2.0
609   609    0.0
610 rows × 2 columns
```

```
sample_submission.to_csv('sample_submission.csv', index=False)
```

INSTRUCTIONS

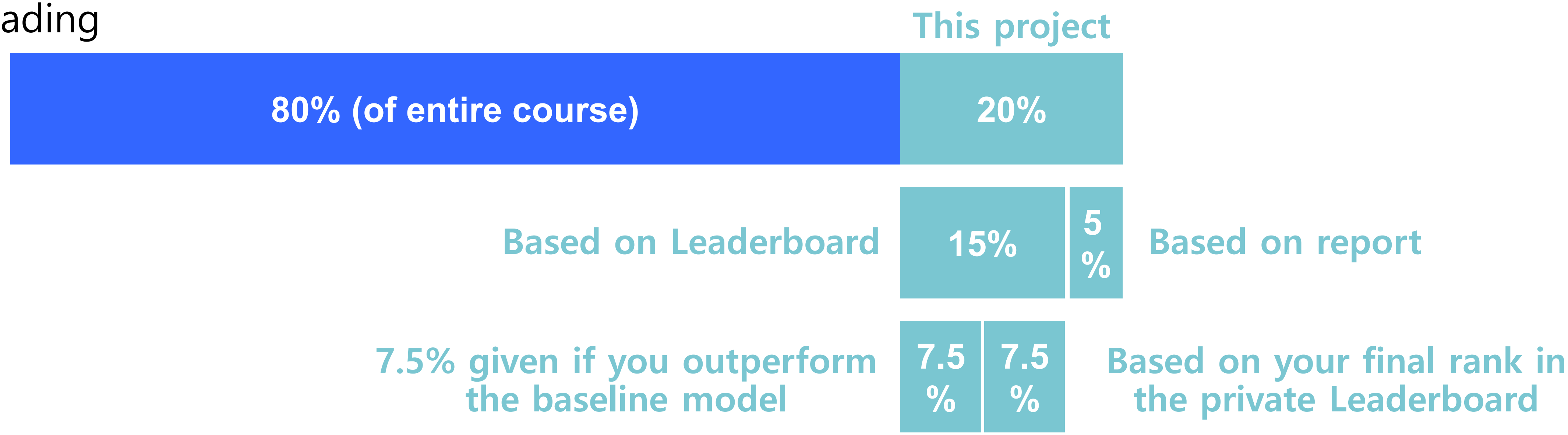
- Submission on Kaggle **(~ 5/27 23:59)**
  - Enter the competition through the link in page 3 (you need a valid email account for registering / signing up)
  - Click on **'Submit Predictions'** in the top right corner
  - Upload your submission file (you may add descriptions to help distinguish between your many trials)
  - If no errors occur and your submission is successful, you can check **your score and ranking on the Leaderboard**
  - Previous submission can by found in 'My Submissions'

  - You can make up to **15 submissions per day**, so we strongly recommend you **start early** and make sure you have enough time

  - You can choose **two** submitted files to use as your final submissions. The system will automatically choose the best one between them once the competition is over.

INSTRUCTIONS

- Grading



This project

80% (of entire course)

20%

Based on Leaderboard

15%

5 %

Based on report

7.5% given if you outperform the baseline model

7.5 %

7.5 %

Based on your final rank in the private Leaderboard

- The **public Leaderboard**, visible to you during the competition, is evaluated on **50% of the test data**. You can use it to get a sense of how well your model performs compared to your classmates.
- The other 50% is used for the **private Leaderboard**, which will be shown to you AFTER the competition ends. The rankings here may be different from the public one. Your ranking here will be your **final result.**

INSTRUCTIONS

- Submission on KLMS **(~ 6/3 23:59)**
  - Code: Please submit a **Jupyter notebook file titled ’20xxxxxx_YourName.ipynb’**. This is the code that reflects your best model. The code is recommended to be well-documented and easy to follow.

  - Report: Please submit a **PDF file titled ’20xxxxxx_YourName.pdf’**. There is no specific format or length requirement, but a detailed explanation on your model should be included. Your ideas and results on analyzing and engineering the data should be detailed as well. Moreover, it is possible to get a **bonus point if you provide 1) anything interesting from this project, 2) further idea to improve the performance.**
    - On the first page, please include a **screenshot of the private Leaderboard, and indicate your username, ranking and score**

INSTRUCTIONS

- This is an **individual project**; sharing your code with classmates is strictly prohibited

- **Do not** enter with more than one account

- If you have any problems or questions regarding the project, feel free to ask through **CLASSUM**

- **Have fun!**