1 PROJECT OBJECTIVES

Students can handle real-world problems by using data coming from the Internet. First, students will formulate a problem. Next, they will apply the knowledge acquired from this course to obtain thoughtful insights, from manually collecting and preprocessing the real-world data, exploring the data with meaningful questions to modelling it using machine learning algorithms. Finally, students use the mined knowledge to solve the formulated problem.

2 PROJECT DESCRIPTION

Students will work in groups to formulate a potential problem which is related to data science and solvable by using real-world data. Instructors will not limit the domains of problems as long as the procedure for solving the formulated problem meets the following requirements:

2.1 Data collection

Each group must collect data related to the topic that needs to be solved. Students must manually manage and process the data using the supporting tools introduced in the lectures, e.g., by using selenium, request, or API. Using published or legacy datasets is not allowable in this project. The data should be structuralized into a table of at least five attributes and 1000 records.

2.2 Data explorations (usually interleaved with the data preprocessing phase)

Students investigate the collected data using descriptive statistics to understand the data better, i.e., to identify data problems (data with missing values, invalid values, columns with data types that are not suitable for further processing, etc.). Here is some information to look into:

- What is the meaning of each column?
- What is the current data type of each column? Are there columns having inappropriate data types?
- With each column, how are values distributed?

2.3 Asking meaningful questions that need to be answered

Each group needs to give at least **five meaningful questions** which can be answered with this data. All the questions should be meaningful (What are the benefits of finding the answer?), and you can not answer them explicitly.

In the notebook file, for each question, each group needs to present:

- What is the question?
- What are benefits of finding the answer?
- How to find the answers of questions by using data

The answers need to be represented by visualization so that instructors and other groups can understand them with no doubt.

2.4 Data modelling and model evaluation

Depending on the problem, students model the data by using machine learning algorithms such as regression algorithms, classification algorithms, and so on. Analyzing and selecting highly relevant and essential features of the problem are also necessary.
For examples:

- House prices prediction using house features (number of bedrooms, area, etc.)
- User classification (trusted, untrusted) based on user features (number of purchase days, monthly/quarterly/yearly purchase amount, and so on).

After choosing the appropriate modelling algorithms, each group must thoroughly validate the model's hyperparameters using data science techniques (e.g., a validation set or cross-validation technique) and report the fine-tuning process. The models' performances must be benchmarked using common classification metrics (precision, accuracy, and recall) or regression metrics (mean square error and root mean square error).
Students are highly recommended to benchmark many modelling algorithms and report the comparison of multiple algorithms.

2.5 Reflection

Once the project is completed, each group composes a report to re-evaluate the work as follows:

- Each member: What difficulties have you encountered?
- Each member: What have you learned?
- Your group: What would you do if you had more time?

3 GROUP WORKING

Each group must use Git and GitHub to do version control and interact with other members effectively. Every phase or task must have its own branch instead of committing everything to the master branch. Each group needs to make sure that:

- A plan for each task is made carefully (Who will do the task? How long will it take to solve it?)
- The amount of work is balanced between members (Commit history in Github should show that)
- Members must understand the work of teammates thoroughly.

The plan and the schedule should be monitored using tools like Notion and Trello. Each group needs to show the overall strategy and each member's work in the final report slide.

4 PROJECT ORGANIZATION

The requirement of organization and implementation of the project:

- The folder organization: Notebook files should be clearly separated for each stage, from data collection, preprocessing, analysis to modeling.
- Answering questions should be demonstrated through the visualization figures and student's contemplative explanations.
- There must be an explanation for every code cell in the Jupyter notebooks. Those containing codes only will be ignored.

5 SUBMISSION AND FINAL SEMINAR

Each team will set up a GitHub repository in a private setting. The repository will be public the day before the seminar so that instructors and selected individuals can review all the work. It is important to note that the grade will suffer negatively if the team pays little effort (e.g., less than ten GitHub commits) or uses tricks on the last days (e.g., making all the commits go to a few days).

Submission files for the final version on Moodle include the following:

- All Jupyter notebooks (and optionally, Python source codes)
- Presentation slide
- Data for the final project (link to Google Drive, One Drive, etc.)

On the presentation day, each group will have less than 15 minutes to present (The teaching assistant will decide the order of presenters) and 10 minutes for Q & A. Therefore, students should practice ensuring the presentation fits the given duration. Furthermore, the presentation should focus on work clearly, instead of solely on code.

The project will be graded zero if there is any dishonest behaviour detected. You can use online documents for reference, but you must provide complete citations. You are free to discuss your topics with classmates, but your work must be implemented and interpreted according to your own understanding.

6 CRITERIA FOR GRADING

Criteria for grading	Proportion
Apply suitable data science processes (collection, preprocessing, analysis, data modelling)	40%
Create questions and derive important, useful insights from the data	20%
Implement and explicitly explain the development process	15%
Compare to other modelling methods and point out the advantages and disadvantages	5%
Presentation	10%
Questions and Answers	10%
Bonus (interesting problem, impressive solution or further studies)	10%
Total	**110%**

The criteria will scale according to the difficulty of each group's project.

7 Contact

If you have any further questions for this project, contact the instructors at:

Lê Nhựt Nam : lnnam@fit.hcmus.edu.vn

Instructors will answer your questions as soon as possible.