

NTHU Introduction to ML 2024

Lab 2

Mortality Prediction using Decision Tree ***and Random Forest***

Ya-Ting Lin

Po-Chih Kuo

Introduction

- Machine learning is playing an increasingly important role in healthcare, where its ability to analyze vast amounts of data can directly **impact human lives**.
- By implementing predictive models on ICU patient data, you'll explore how machine learning can be used to support critical healthcare decisions. Whether it's identifying at-risk patients or improving care outcomes, these techniques offer the potential to transform patient care and **save lives**.
- By the end of this lab, you will have a deeper understanding of how machine learning can be applied to healthcare scenarios, where every decision can make a difference.

Dataset

- **Real** Data
 - A database containing a large amount of critical care data from many different intensive care units (ICUs) worldwide
- Basic Part: We extracted 40 cases with 10 attributes and 1 label ('hospital_death')
- Advanced Part: We extracted 8500 cases with 29 attributes and 1 label ('hospital_death')

Goal

- Be familiar with the concepts of building a decision tree
- Implement a decision tree
- Implement a random forest
- Make predictions on patients' survival ('hospital_death') from real data
- Fine-tune the model for better performance

You will have the following items



- Template : lab2.ipynb
- Input file :
 - lab2_basic_input.csv
 - lab2_advanced_training.csv
 - lab2_advanced_testing.csv (without label 'hospital_death')

Template

- You must use the given file **lab2.ipynb** to build the model
- Except for the imported packages in the template, you **cannot** use any other packages
- Please follow the template, and only modify the content where we specifically indicate you can.

Mount Google Drive (optional)

```
[ ] from google.colab import drive
drive.mount('/content/drive')
```

▼ **Lab 2 : Decision Tree and Random Forest**

In *lab 2*, you need to finish :

1. Basic Part : Implement a Decision Tree model and predict whether patients in the validation set survived.
 - Section 1: Function Implementation and Testing
 - Section 2: Building the Decision Tree Model
2. Advanced Part : Build a **Random Forest** model to make predictions

! Important ! Please follow the template. Follow the instructions. **Do not** change the code outside this code bracket if you see one.

```
### START CODE HERE ###
...
### END CODE HERE ###
```

We'll be using **pandas** frequently in this template, so we've provided a link to help you get familiar with its usage:
https://pandas.pydata.org/docs/user_guide/10min.html

Basic Input File Format

- Named “lab2_basic_input.csv”
 - 40 instances in total
 - Each instance has 10 features and 1 class label

	age	bmi	gender	height	weight	pre_icu_los_days	glucose_apache	heart_rate_apache	resprate_apache	sodium_apache	hospital_death
0	28.0	26.596278	1	173.0	79.60	0.000000	199.0	52.0	29.0	140.0	0
1	51.0	36.267895	0	180.3	117.90	0.141667	88.0	104.0	31.0	143.0	0
2	81.0	24.196007	1	162.0	63.50	1.988194	285.0	178.0	4.0	138.0	1
3	83.0	21.105377	1	162.6	55.80	0.211111	189.0	115.0	18.0	158.0	0
4	76.0	20.470093	0	167.6	57.50	14.493056	278.0	93.0	8.0	134.0	1
5	60.0	46.111111	0	180.0	149.40	0.027778	186.0	146.0	34.0	139.0	1
6	70.0	17.361111	1	168.0	49.00	0.156944	181.0	111.0	12.0	158.0	1



Advanced Training File Format

- Named “*lab2_advanced_training.csv*”
 - 8500 instances in total
 - Each instance has 29 features and 1 class label

29 features

Class label

	age	bmi	gender	height	weight	pre_icu_los_days	arf_apache	bun_apache	creatinine_apache	gcs_eyes_apache	...	aids	cirrhosis	diabetes_mellitus	leukemia	hospital_death
0	79.0	25.616497	1	168.0	72.3	0.305556	0.0	20.0	0.92	4.0	...	0.0	0.0	0.0	0.0	0
1	43.0	23.494409	0	171.0	68.7	0.011806	0.0	9.0	0.70	1.0	...	0.0	0.0	0.0	0.0	0
2	62.0	29.145882	0	182.9	97.5	0.006250	0.0	54.0	3.59	1.0	...	0.0	0.0	0.0	0.0	1
3	72.0	41.183318	1	170.2	119.3	1.945139	0.0	53.0	2.25	4.0	...	0.0	0.0	1.0	0.0	1
4	87.0	22.914211	0	170.1	66.3	0.085417	0.0	33.0	1.60	4.0	...	0.0	0.0	0.0	0.0	0

Advanced Testing File Format

- Named “***lab2_advanced_testing.csv***”
 - 900 instances in total
 - Each instance has 29 features
 - Without class label

29 features

	age	bmi	gender	height	weight	pre_icu_los_days	arf_apache	bun_apache	creatinine_apache	gcs_eyes_apache	...	sodium_apache	temp_apache	ventilated_apache	wbc_apache
0	82	38.733847	1	158.23	96.82	0.232639	0.0	50	3.32	1.0	...	135	33.0	1.0	14.8
1	65	22.692476	0	173.67	69.40	0.121528	0.0	33	1.40	1.0	...	133	32.1	1.0	12.5
2	72	33.702285	0	177.47	105.70	0.143750	0.0	17	1.71	1.0	...	143	33.9	1.0	17.8
3	81	20.274075	0	171.74	61.10	0.664583	0.0	35	2.09	3.0	...	136	36.4	1.0	9.0
4	41	29.027749	1	175.75	90.00	0.004167	0.0	3	0.41	1.0	...	149	32.3	1.0	24.0

Grading Policy



Item	Score
Basic Implementation (Decision Tree)	30%
Advanced Implementation (Random Forest)	65%
Report	5%

Basic Implementation (30%)

- **Section 1: Function Implementation and Testing**
 - Implement 5 functions that are necessary for building a decision tree model.
 - After implementing each function, you must run it with the given input variables to verify its correctness.
- **Section 2: Build a Decision Tree Model and make Predictions**
 - Use the functions from section 1 to build a decision tree model and make predictions.
- Please use *lab2_basic_input.csv* as your input data

Basic Grading Policy



- Given information on 40 patients and whether they survived
- Section 1: Function Implementation and Testing
 - Step 1 : Calculate the Entropy
(5%)
 - Step 2 : Calculate the Information Gain
(5%)
 - Step 3 : Find the Best Split
(5%)
 - Step 4 : Split the data into two branches
(5%)
 - Step 5 : Build the decision tree
(5%)
 - Step 6: Save answers
- Section 2: Build a Decision Tree Model and make Predictions
 - Step 1: Split the data into training set and validation set
 - Step 2: Train a decision tree model with the training set
 - Step 3: Predict the cases in the validation set by using the model trained in Step 2
 - Step 4: Calculate the f1-score of your predictions in Step 3
(5%)



Basic Output File Format

- Please save your answers into **lab2_basic.csv**
 - Submit the file to **eeclab**
- There should be 7 rows in your csv file:

row number	description	variable
Row 1	Header	['Id', 'Ans']
Row 2	entropy	'ans_entropy'
Row 3	information gain	'ans_informationGain'
Row 4	best split information gain, value, feature	['ans_ig', 'ans_value', 'ans_name']
Row 5	number of instances in the left subtree	'ans_left'
Row 6	n features and the threshold corresponding to each feature	'ans_features' + 'ans_thresholds'
Row 7	F1-score	'ans_f1score'

Advanced Implementation (65%)

- Build a random forest
- Please use *lab2_advanced_training.csv* as the training data
- Make predictions with the Random Forest on the testing data *lab2_advanced_testing.csv*

Advanced Grading Policy

- Baseline – 55%
 - F1-Score ≥ 0.65 (25%)
 - F1-Score ≥ 0.68 (15%)
 - F1-Score ≥ 0.7 (15%)
- Ranking – 10%
 - Compete your F1-Score with the whole class



Advanced Output File Format

- There should be (900+1) rows in your csv file
 - First row is the header ['Id', 'hospital_death']
 - Your prediction answer should be either 0 or 1
 - Id starts from 0, and **hospital_death** is the predicted answer
- Please make sure that your output format is correct
- Submit the answer (.csv) to Kaggle **ML2024-Lab2-AdvancedPart**

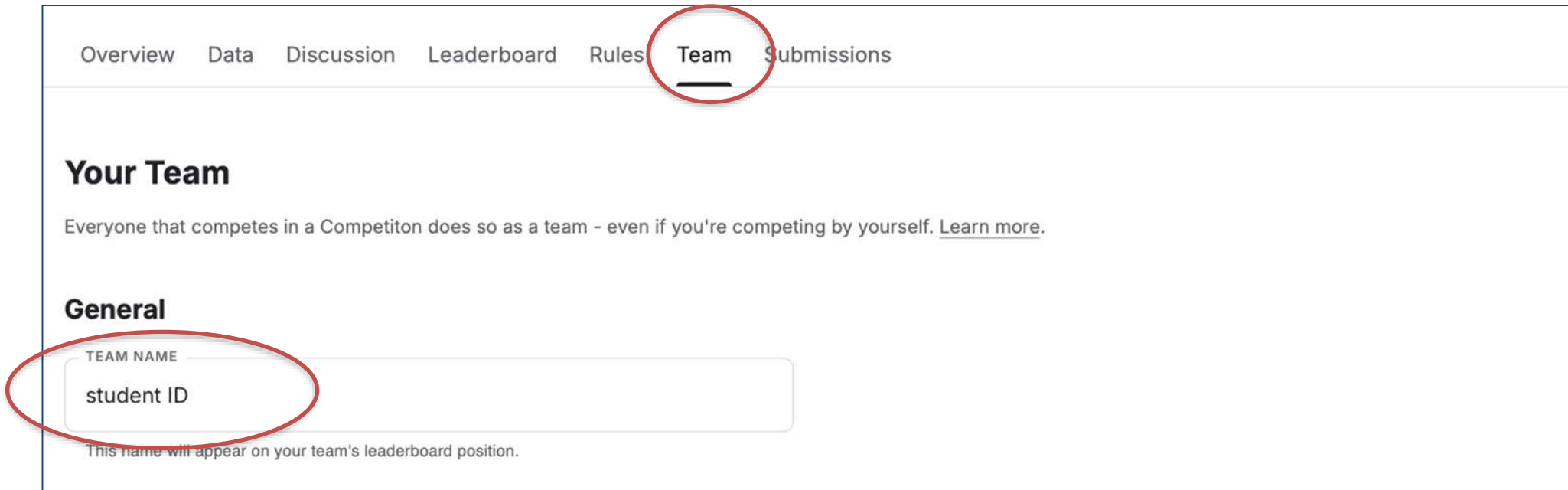
Id	hospital_death
0	1
1	1
2	1
3	1
4	1
5	0
6	0
7	1
8	1
9	1
10	0
11	0
12	0
13	1
14	0

Kaggle

- We've created a competition for the Advanced part
- link: <https://www.kaggle.com/t/f429abd0842e414e9685155f9bcb21ce>
 - In the advanced part, we split the testing data into **public** & **private** parts.
 - The score you see on kaggle after submission is your public score
 - You can directly check if you have passed the three baselines
 - Private score is for ranking.
 - The private score will be revealed after the deadline.

Kaggle

- After joining the competition, you should change your team name (each student is a team) to your **student ID**.



The screenshot shows the 'Your Team' page on Kaggle. The 'Team' tab in the top navigation bar is circled in red. Below the 'General' section, the 'TEAM NAME' input field is also circled in red and contains the text 'student ID'. A note below the input field states: 'This name will appear on your team's leaderboard position.'

Overview Data Discussion Leaderboard Rules **Team** Submissions

Your Team

Everyone that competes in a Competition does so as a team - even if you're competing by yourself. [Learn more.](#)

General

TEAM NAME

student ID

This name will appear on your team's leaderboard position.

Report

- Named as “**lab2_report.pdf**”
- Briefly describe the attributes setting of the random forest model , including:
 - The number of trees you used (1%)
 - The number of features you used (1%)
 - The number of instances you used to build each tree (1%)
 - (optional) any other settings
- Briefly describe the difficulty you encountered (1%)
- Summarize how you solved the difficulty and your reflections (1%)
- **No more than one page**

Lab 2 Requirements

- Do it individually! Not as a team! (The team is for the final project)
- Announce date: 2024/10/1
- Deadline: 2024/10/15 23:59 (Late submission is not allowed!)
- Hand in your files in the following format (Do not compress!)
 - lab2.ipynb
 - lab2_report.pdf
 - lab2_basic.csv

The Evaluation Metric

- F1-score

$$F1\text{-score} = 2 \times \frac{(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})}$$

- For example
 - The class you predicted:
 $\hat{y} = [1, 1, 0, 0, 0, 0, 1]$
 - Actual values:
 $y = [0, 0, 0, 0, 0, 1, 1]$
 - F1-score = 0.4

		Actual/True value	
		positive	negative
Pre dic ted val ue	posi tive	TP	FP
	neg ativ e	FN	TN

		Actual/True value	
		positive	negative
Pre dic ted val ue	posi tive	TP	FP
	neg ativ e	FN	TN



Penalty

- 0 points if any of the following conditions happened
 - Plagiarism
 - Late submission
 - Not using a template or importing any other packages in this assignment
 - No submission record on Kaggle
 - Your submission was not generated by your code
 - Not following the instructions to print certain answers in the template
 - Kaggle's team name is not your student ID (we cannot identify who you are)

Questions?

- TA: Ya-Ting Lin (ivylin752@gmail.com)
- Do not ask for debugging.
- **TA time for 10/3 and 10/9 will be moved to 17:30~18:30**

