Assignment 2

Diabetes Prediction using Decision Tree and Random Forest

Bao-Hsuan Huang

Po-Chih Kuo

Introduction

- Diabetes mellitus is a common chronic disease, and it may cause many complications. According to statistics, the morbidity of diabetes has been on the rise in recent years.
- In about 20 years, the world's diabetic patients will reach 642 million, which means that one in every ten adults will have diabetes in the future.
- Therefore, in this assignment, we need to analyze the given ICU dataset and predict whether the patient suffers from diabetes.

Dataset

- GOSSIS dataset (The Global Open-Source Severity of Illness Score)
- ***Real*** Data
  - The real data collected by GOSSIS consortium
  - A database contains a large amount of critical care data from many different intensive care units (ICUs) worldwide
- Basic Part : We extract 30 cases with 9 attributes and 1 label (‘diabetes_mellitus’)
- Advanced Part : We extract 8379 cases with 24 attributes and 1 label

Goal

- Implement a decision tree with *GOSSIS* dataset
- Implement a random forest by using your decision tree model
- Predict the patients' diabetes ('diabetes_mellitus') from real data
- Fine-tune the model for better performance

Grading Policy

Item	Score
Basic Implementation (Decision Tree)	60%
Advanced Implementation (Random Forest)	35%
Report	5%

Basic Implementation (60%)

- Given information on several patients and whether they have diabetes
- Build a decision tree in following steps with *diabetes detection* dataset
  - Step 1 : calculate the entropy (10%)
  - Step 2 : calculate the information gain (10%)
  - Step 3 : search for the best split (10%)
  - Step 4 : split data into 2 branches (10%)
  - Step 5 : build the decision tree (10%)
  - Step 6 : make predictions by decision tree (10%)
- Please use ***hw2_input_basic.csv*** as your input data
- You **don't** need to use ***hw2_input_test.csv*** in the basic part
- Please save your answer in ***hw2_basic.csv***

Advanced Implementation (35%)

- Build a random forest by using at least **3** decision trees
- Please use ***hw2_input_advanced.csv*** as the input data
- Please use ***hw2_input_test.csv*** as the test data and make the predictions
- Make predictions with the test data
  - Please save the predictions in ***hw2_advanced.csv***

Advanced Grading Policy

- Make predictions with Random Forest on the test data in ***hw2_input_test.csv***
- Baseline – 20%
  - F1-Score >= 0.55
- Ranking – 15%
  - We will calculate F1-Score to compete with the whole class

You will have the following items

- Template : hw2.ipynb
- Input file :
  - hw2_input_basic.csv
  - hw2_input_advanced.csv
  - hw2_input_test.csv (without label data)
- Sample output file :
  - sample_basic.csv
  - sample_advanced.csv

Template

- You must use the given file **hw2.ipynb** to build the model
- Except for the imported packages in the template, you **cannot** use any other packages

HW2 : Decision Tree and Random Forest

In *assignment 2*, you need to finish :

1. Basic Part : Implement a **Decision Tree** model and predict whether the patients in the validation set have diabetes
    - Step 1 : Load the input data
    - Step 2 : Calculate the Entropy and Information Gain
    - Step 3 : Find the Best Split
    - Step 4 : Split into 2 branches
    - Step 5 : Build decision tree
    - Step 6 : Save the answers from step2 to step5
    - Step 7 : Split data into training set and validation set
    - Step 8 : Train a decision tree model with training set
    - Step 9 : Predict the cases in the *validation set* by using the model trained in *Step8*
    - Step 10 : Calculate the f1-score of your predictions in *Step9*
    - Step 11 : Write the Output File
2. Advanced Part : Build a **Random Forest** model to make predictions
    - Step 1 : Load the input data
    - Step 2 : Load the test data
    - Step 3 : Build a random forest
    - Step 4 : Predict the cases in the test data by using the model trained in *Step3*
    - Step 5 : Save the predictions(from *Step 4*) in a csv file

Basic Input File Format

- Named "**hw2_input_basic.csv**"
  - 30 instances in total
  - Each instance has 9 features and 1 class label

9 features

Class label

age	bmi	gender	height	weight	glucose_apache	heart_rate_apache	resprate_apache	sodium_apache	diabetes_mellitus
70	25.98465933	1	172.7	77.5	116	101	49	137	0
30	31.31036825	1	170.2	90.7	71	39	33	144	0
54	24.38882429	1	177.8	77.1	120	120	31	141	0
65	34.14107409	0	170.2	98.9	73	48	36	140	1
49	22.56474287	1	172.7	67.3	207	119	6	144	0
62	29.42401041	0	154.9	70.6	113	60	32	137	0
85	27.67357353	1	154.9	66.4	102	49	36	142	0
65	22.26943229	1	177.8	70.4	333	59	6	145	1

Advanced Input File Format

- Named "**hw2_input_advanced.csv**"
  - 8379 instances in total
  - Each instance has 24 features and 1 class label

24 features

Class label

age	bmi	gender	height	.....	.....	apache_4a_hospital	apache_4a_icu_dea	diabetes_mellitus
72	35.02716161	1	188			0.2	0.12	1
68	23.99402733	1	180.3			0.05	0.02	1
54	29.56654595	1	188			0.06	0.04	1
42	16.26190759	1	182.9			0.01	0	1
82	24.01776785	0	162.6			0.07	0.03	0
42	33.26036394	1	172.7			0.02	0.01	0
73	28.12148481	1	177.8			0.02	0.01	0
64	27.36810207	0	165.1			0.42	0.31	0

Advanced Input Test File Format

- Named "**hw2_input_test.csv**"
  - 840 instances in total
  - Each instance has 24 features
  - Without class label

24 features

age	bmi	gender	height	.....	ventilated_apache	wbc_apache	apache_4a_hospital	apache_4a_icu_dea
62	32.86639226	1	177.8	.....	0	4.56	0.06	0.03
82	23.58276644	0	157.5	.....	0	6	0.14	0.06
61	31.68452008	1	172.7	.....	0	8.59	0.05	0.03
58	45.15625	0	160	.....	1	16.03	0.33	0.22
74	25.81701636	1	172.7	.....	0	45.8	0.12	0.05
19	22.95871667	0	162.6	.....	0	10.6	0.01	0
45	28.11651131	0	162.56	.....	0	5.6	0.01	0

Basic Output File Format

- Named as **hw2_basic.csv**
- There should be (7+2n) rows in your csv file:

row number	description	variable
Row 1	entropy	‘ans_entropy’
Row 2	information gain	‘ans_informationGain’
Row 3~5	best split information gain, value, feature	‘ans_ig’, ‘ans_value’, ‘ans_name’
Row 6	number of instances in the left subtree	‘ans_left’
Row 7 ~ 7+(n-1)	n features you used	‘ans_features’
Row 7+n, 7+(2n-1)	the threshold corresponding to each feature	‘ans_thresholds‘
Row 7+2n	F1-score	‘ans_f1score’

Basic Output File Format

- Example:

1	0.9709505916	→ ans_entropy
2	0.0473011807	→ ans_informationGain
3	0.2812908988	ans_ig
4	51.5	ans_value
5	age	ans_name
6	10	→ ans_left
7	resprate_apache	ans_features
8	age	
9	27	ans_thresholds
10	51.5	
11	0.5416666667	→ ans_f1score

- Please make sure that your output format is correct
  - You can refer to the output format of **sample_basic.csv**

Advanced Output File Format

- Named as **hw2_advanced.csv**
- y_test contains 840 instances
- There should be 840 rows in your csv file
  - Without header
  - Your prediction answer should be either 0 or 1
- Please make sure that your output format is correct
  - You can refer to the output format of **sample_advanced.csv**

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

Report

- Named as “**hw2_report.pdf**”
- Briefly describe the attributes setting of the random forest model (2%), including:
  - The number of trees you used
  - The number of features you used
  - The number of instances you used to build each tree
  - (optional) any other settings
- Briefly describe the difficulty you encountered (1%)
- Summarize how you solve the difficulty and your reflections (2%)
- No more than one page

Assignment 2 Requirement

- Do it individually! Not as a team! (The team is for final project)
- Announce date: 2022/10/20
- Deadline: 2022/11/2 23:59 (Late submission is not allowed!)
- Hand in your files in the following format (Do not compressed!)
    - hw2_basic.csv
    - hw2_advanced.csv
    - hw2.ipynb
    - hw2_report.pdf
- Assignment 2 would be covered on the exam next time

The Evaluation Metric

- F1-score

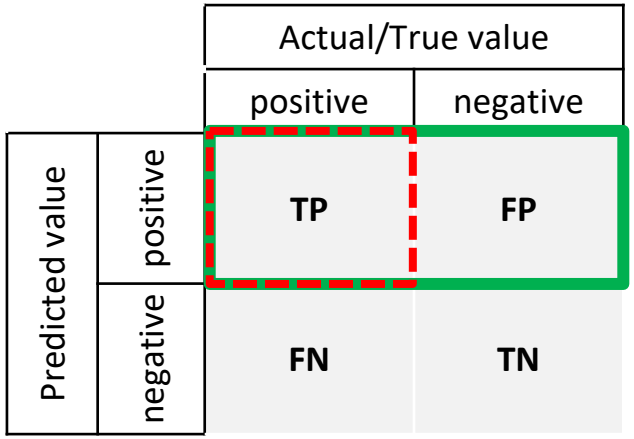
$$F1\text{-}score = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)}$$

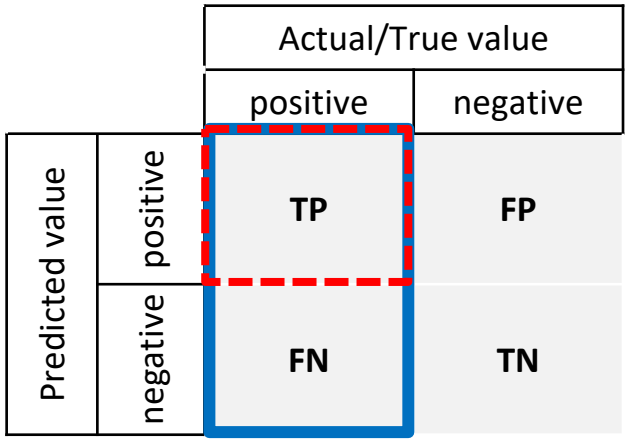
- For example
  - The class you predicted:

    $\hat{y}$ = [1, 1, 0, 0, 0, 0, 1]
  - Actual values:

    y = [0, 0, 0, 0, 0, 1, 1]

  - F1-score = 0.4





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
  - Incorrect prediction format
  - Incorrect submission format

Questions?

- TA: Bao-Hsuan Huang (thebhhuang@gmail.com)
- Do not ask for debugging.