CMU 10-742 (Fall 2024) - Machine Learning in Healthcare

Assignment 2: Predicting Sepsis

Out: Thurs Sep 12

Due: Thurs Sep 26

This assignment counts for 9 points out of the 35 total points allocated to the course problem sets.

In this assignment, we'll first familiarize ourselves with MIMIC data. Then, in Part 2, we'll build a model to predict which patients in an intensive care unit (ICU) will develop sepsis. In Part 3, we'll go through a brief refresher on Bayesian statistics.

Part 1: EDA (2 points)

In this part, we'll conduct some exploratory data analysis (EDA) on several MIMIC datasets, to familiarize ourselves with healthcare codes and some idiosyncracies in the data.

```
# collecting (most) of the imports for this assignment in one place. You may not end
# you may need others not listed here.

from google.colab import auth
import pandas as pd
import numpy as np
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, roc_auc_score, roc_curve, confusion_matr
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt
from sklearn.feature_extraction.text import CountVectorizer


auth.authenticate_user()
```

Download these files from Physionet to your computer, uncompress them, and then upload the resulting files here (colab). There is a file icon on the left panel of colab - you can use this to upload files from your computer to colab.

`https://physionet.org/content/mimiciii/1.4/DIAGNOSES_ICD.csv.gz` to `MIMIC_III_DIAGNOSES_ICD.csv` `https://physionet.org/content/mimiciii/1.4/D_ICD_DIAGNOSES.csv.gz` to `MIMIC_III_D_ICD_DIAGNOSES.csv`

`https://physionet.org/files/mimiciv/2.2/hosp/d_icd_diagnoses.csv.gz` to `MIMIC_IV_D_ICD_DIAGNOSES.csv`

`https://physionet.org/files/mimiciv/2.2/hosp/diagnoses_icd.csv.gz` to `MIMIC_IV_DIAGNOSES_ICD.csv`

```
# In case physionet server is unresponsive, we've placed the required MIMIC files
# in a private GCP bucket. We will only grant access to this bucket (a) in case
# the physionet server is down, and (b) only to those students who have proven
# to the course staff that they have been granted access to the MIMIC III and IV
# file repositories on physionet.

#bucket="10-742-mimic"
#!gsutil cp gs://{bucket}/MIMIC_III_DIAGNOSES_ICD.csv ./
#!gsutil cp gs://{bucket}/MIMIC_III_D_ICD_DIAGNOSES.csv ./
#!gsutil cp gs://{bucket}/MIMIC_IV_D_ICD_DIAGNOSES.csv ./
#gsutil cp gs://{bucket}/MIMIC_IV_DIAGNOSES_ICD.csv ./


# Load the ICD 'code to name' dictionaries

icd9_code_to_names = pd.read_csv('MIMIC_III_D_ICD_DIAGNOSES.csv')
mimic_iv_icd_to_name = pd.read_csv('MIMIC_IV_D_ICD_DIAGNOSES.csv')

# Create a mapping of ICD9 codes to names
icd9_to_name = dict(zip(icd9_code_to_names.ICD9_CODE, icd9_code_to_names.LONG_TITLE)

# Do the same for ICD10
icd10_only = mimic_iv_icd_to_name[mimic_iv_icd_to_name['icd_version'] == 10]
icd10_to_name = dict(zip(icd10_only.icd_code, icd10_only.long_title))

#uncomment if you're curious
#print ("Here's the first few items in the 'icd9_to_name' dict:")
#for key, value in list(icd9_to_name.items())[:10]:
#    print(f"Key: {key}, Value: {value}")

#print ("Here's the first few items in the 'icd10_to_name' dict:")
#for key, value in list(icd10_to_name.items())[:10]:
#    print(f"Key: {key}, Value: {value}")
```

1.1

What are the top ten most common ICD-9 codes in MIMIC-III and what percent of total ICD-9 codes does each ICD-9 code in the top ten make up (e.g, Hypertension, 1%)? Please provide the description of the ICD code and the code itself. You can use the two dictionaries we loaded in the previous step: `icd9_to_name` and `icd10_to_name`.

YOUR ANSWER HERE

1.2

What are the top ten most common ICD-10 codes in MIMIC-IV and what percent of total ICD-10 codes does each ICD-10 code in the top ten make up (e.g, Hypertension, 1%)? Provide the description of the ICD code and the code itself.

Note that MIMIC-IV has both ICD-9 and ICD-10 codes, since the MIMIC-IV dataset was collected before and after the cutover in ICD version. For MIMIC-IV, you should only consider the rows where `icd_version="10"`.

YOUR ANSWER HERE

1.3

What is the average number of ICD codes per visit in MIMIC-III?

Hint: The `HADM_ID` field is a unique identifier for each hospital admission of a patient.

YOUR ANSWER HERE

1.4

Same as previous question, but for MIMIC-IV. Once again, only consider ICD-10 codes for this dataset.

YOUR ANSWER HERE

1.5

Plot the distribution of the number of admissions for each patient in MIMIC-IV (as always with MIMIC-IV, we only want you to consider ICD-10 codes).

Use a logarithmic scale for the y-axis (why?).

YOUR ANSWER HERE

1.6

Graph the distribution of number of ICD-10 codes per enounter in the MIMIC-IV dataset. What do you notice at the outermost edge of the distribution? Offer up a plausible explanation.

YOUR ANSWER HERE

1.7

Inspect the list of top outpatient ICD-10 codes in 2021 from this url:
https://www.definitivehc.com/blog/top-outpatient-diagnoses-by-icd-10-code

How does it differ from the top 10 ICD-10 codes you found in MIMIC-IV? Can you offer any explanation?

YOUR ANSWER HERE

Part 2: Predicting Sepsis (5 points)

Sepsis is a serious and unfortunately common issue amongst hospitalized patients, contributing to 6 million deaths per year. Sepsis occurs when an infection causes a systemic inflammatory response, disrupting normal physiologic functioning. This can lead to septic shock, a situation in which the body cannot maintain proper blood pressure, resulting in inadequate blood flow to the organs, depriving them of the oxygen and nutrients they need to function properly. Early deployment of broad-spectrum antibiotics and fluid resuscitation can save lives...and the earlier, the better.

In this part, we will use the dataset originally published for the 2019 PhysioNet Computing in Cardiology Challenge. The dataset was assembled from over 60K patients in ICUs from three separate hospital systems. The data contains up to 40 clinical variables for each hour of a patient's ICU stay. The documentation for this dataset is here, but we have made the following modifications to the data for this assignment:

- The `SepsisLabel` field stores the hour after admission when the patient developed sepsis, or 0 if they did not develop sepsis.
- We retain only the first 5 time hours of data after admission
- We discard all records where the patient stayed for fewer than 8 hours or developed sepsis earlier than 8 hours after admission.

```
# download data from course cloud folder. For reference, the original source of this
# https://archive.physionet.org/users/shared/challenge-2019/. It contains data from

!gsutil cp gs://10-742/assignment_2/asst2_train.zip ./
!gsutil cp gs://10-742/assignment_2/asst2_test.zip ./


# Load data into a set of dataframes, one per patient.
# The number of rows in a dataframe is the number of events for that patient.
import zipfile
from tqdm import tqdm

def read_dataframes(zip_path, max_files=100):
    dataframes = []
    with zipfile.ZipFile(zip_path, 'r') as zf:
        file_list = zf.namelist()[:max_files]  # Limit to the first `max_files` file
        for file_name in tqdm(file_list, desc="Reading DataFrames from " + zip_path)
            # Read each CSV file into a DataFrame
            with zf.open(file_name) as file:
                df = pd.read_csv(file)
                dataframes.append(df)
    return dataframes

# Read DataFrames back from the ZIP archive
training_data = read_dataframes("asst2_train.zip", max_files=999999)
test_data = read_dataframes("asst2_test.zip", max_files=999999)

# at this point, training_data and test_data are lists consisting of a dataframe for
print(f"\nThere are {len(training_data)} patients in training set and {len(test_data
print(f"The first patient in the training data has {len(training_data[0])} records."
print(f"The first patient in the test data has {len(test_data[0])} records.")
```

2.1

Carefully inspect the first few readings for one of the patients, so that you understand the data. For fields you don't understand, refer to https://physionet.org/content/challenge-2019/1.0.0/

All the fields are important, but in particular be sure you understand ICULOS.

For this question and all questions up to and including 2.5, you should restrict attention to the training data.

2.2

What is the fraction of patients that eventually develop sepsis? We'll call this the 'sepsis cohort' from now on.

For this through 2.5, use only the training data.

YOUR ANSWER HERE

2.3

Examine the distributions of gender and age between the patients with sepsis and the full set of patients. What differences do you notice? Be sure to show any outputs or plots that illustrate your observations.

YOUR ANSWER HERE

2.4

Plot a histogram of the first hour in which patients develop sepsis. In other words, the number of hours since the patient was admitted to the hospital.

YOUR ANSWER HERE

2.5

There's a LOT of missing data in this dataset. Which is not atypical for a medical dataset.

Calculate how often each feature is missing across all patients (sepsis or not). For example, if patient1 has feature1 missing 2/40 (across the 40 hours of their stay), and patient2 has feature1 missing 5/30, the missingness for feature 1 is (2+5)/(40+30). Show the top k=10 features with the highest "missingness."

YOUR ANSWER HERE

2.6

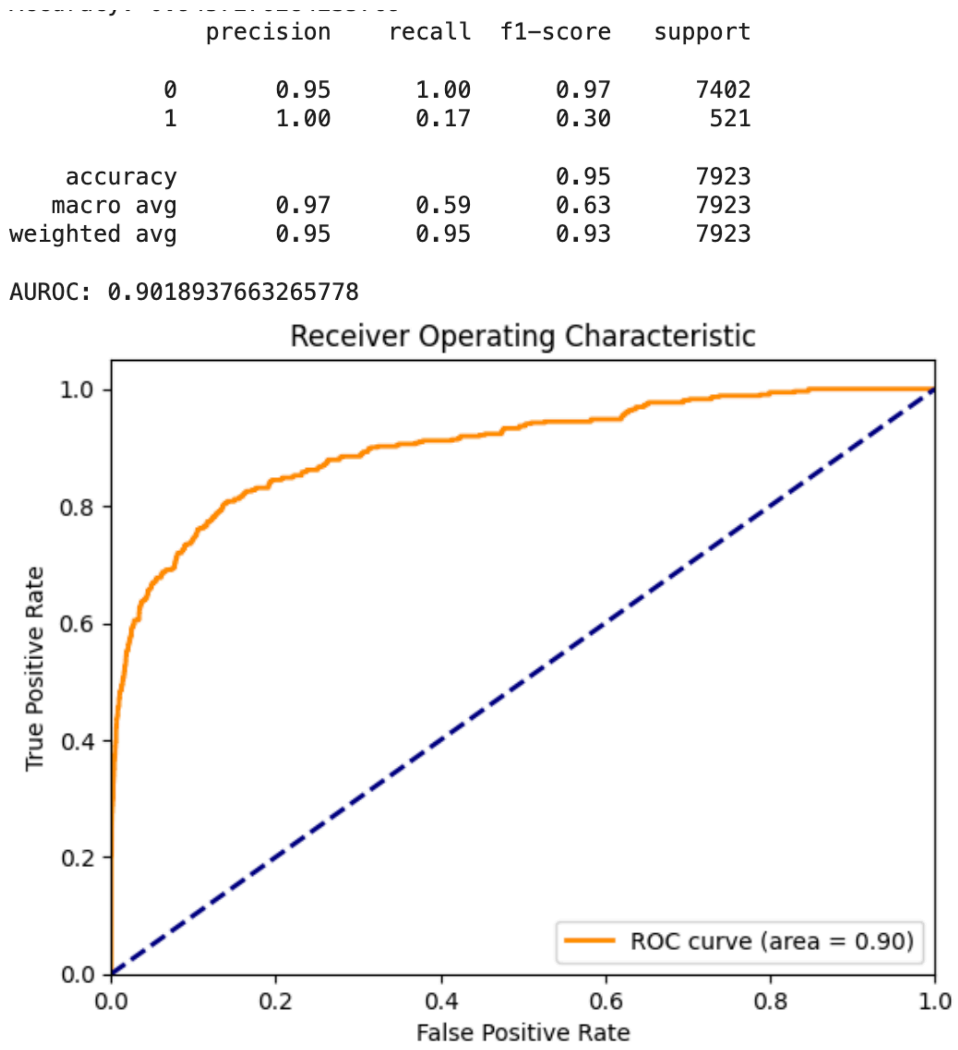
Now we're going to build a model to predict whether a patient develops sepsis. Not *when* they develop sepsis, but *if*.

How you attack the problem is up to you. You decide how to handle the missing data (recall we discussed some imputation methods in lecture 5). You decide what features to extract from the data. You decide what form of model to use, and what hyperparameter settings to use. As you decide on the form of your model, be sure to consider the next question in this assignment.

Succinctly describe your assumptions and approach in a free text cell.

Show your model performance on the test set (`test_data`) as an ROC curve.

For reference, here's what we got:



```
              precision    recall  f1-score   support

           0       0.95      1.00      0.97      7402
           1       1.00      0.17      0.30       521

    accuracy                           0.95      7923
   macro avg       0.97      0.59      0.63      7923
weighted avg       0.95      0.95      0.93      7923

AUROC: 0.9018937663265778
```

SUCCINCTLY DESCRIBE YOUR ASSUMPTIONS AND APPROACH HERE

```
SHOW YOUR CODE AND ROC CURVE HERE
```

2.7

You are presenting this trained model at Springfield Hospital's Grand Rounds next week. You should expect someone in the audience to "explain" your model. Select three of the top (most influential) features and provide an intuitive explanation for these features.

YOUR ANSWER HERE

2.8

In a clinical setting, an early identification of sepsis is more valuable than a later identification. However, classic ML model metrics (e.g. AUROC) don't capture this aspect – a model that predicts sepsis correctly 8 hours in advance of sepsis onset is equivalent in quality to a model that predicts it 3 hours in advance. Describe, in a few sentences, a proposed metric that takes both accuracy and timeliness into account. Describe how you would modify your approach to build a model that optimizes this metric, vs. traditional AUC.

YOUR ANSWER HERE

Part 3: Simpson's Paradox (2 points)

In the mid-1980s, a group of urologists published a study (link) that compared two treatments for kidney stones: open surgery and percutaneous nephrolithotomy (basically a minimally invasive surgery, similar in aspects to laparoscopic surgery).

Here were their results:

	open surgery	**percutaneous nephrolithotomy**
stone diameter <= 2cm	81/87	234/270
stone diameter > 2cm	192/263	55/80

3.1

For each of the stone sizes, which is the more effective treatment?

Now add another row at the bottom of the table, showing the aggregate effect of the treatment cohort and the control cohort. At this aggregate view, does the treatment appear to help or hurt?

Describe the apparent contradiction in these statistics, and also provide a succinct explanation. This statistical phenomenon is called Simpson's paradox.

YOUR ANSWER HERE