C3_W1_Assignment

May 26, 2025

1 Estimating Treatment Effect Using Machine Learning

Welcome to the first assignment of **AI for Medical Treatment**!

You will be using different methods to evaluate the results of a randomized control trial (RCT).

You will learn: - How to analyze data from a randomized control trial using both: - traditional statistical methods - and the more recent machine learning techniques - Interpreting Multivariate Models - Quantifying treatment effect - Calculating baseline risk - Calculating predicted risk reduction - Evaluating Treatment Effect Models - Comparing predicted and empirical risk reductions - Computing C-statistic-for-benefit - Interpreting ML models for Treatment Effect Estimation - Implement T-learner

1.1 Table of Contents

- Section ??
    - Section ??
    - Section ??
    - Section ??

1.2 Packages

We'll first import all the packages that we need for this assignment.

- pandas is what we'll use to manipulate our data
- numpy is a library for mathematical and scientific operations
- matplotlib is a plotting library
- sklearn contains a lot of efficient tools for machine learning and statistical modeling
- random allows us to generate random numbers in python
- lifelines is an open-source library that implements c-statistic
- itertools will help us with hyperparameters searching

1.3 Import Packages

Run the next cell to import all the necessary packages, dependencies and custom util functions.

```
In [ ]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        import sklearn
        import random
        import lifelines
        import itertools
        from public_tests import *

        plt.rcParams['figure.figsize'] = [10, 7]
```

1. Dataset ### 1.1 Why RCT?

In this assignment, we'll be examining data from an RCT, measuring the effect of a particular drug combination on colon cancer. Specifically, we'll be looking the effect of Levamisole and Fluorouracil on patients who have had surgery to remove their colon cancer. After surgery, the curability of the patient depends on the remaining residual cancer. In this study, it was found that this particular drug combination had a clear beneficial effect, when compared with Chemotherapy. ### 1.2 Data Processing In this first section, we will load in the dataset and calculate basic statistics. Run the next cell to load the dataset. We also do some preprocessing to convert categorical features to one-hot representations.

```
In [ ]: data = pd.read_csv("data/levamisole_data.csv", index_col=0)
```

Let's look at our data to familiarize ourselves with the various fields.

```
In [ ]: print(f"Data Dimensions: {data.shape}")
        data.head()
```

Below is a description of all the fields (one-hot means a different field for each level): - sex (binary): 1 if Male, 0 otherwise - age (int): age of patient at start of the study - obstruct (binary): obstruction of colon by tumor - perfor (binary): perforation of colon - adhere (binary): adherence to nearby organs - nodes (int): number of lymphnodes with detectable cancer - node4 (binary): more than 4 positive lymph nodes - outcome (binary): 1 if died within 5 years - TRTMT (binary): treated with levamisole + fluoroucil - differ (one-hot): differentiation of tumor - extent (one-hot): extent of local spread

In particular pay attention to the `TRTMT` and `outcome` columns. Our primary endpoint for our analysis will be the 5-year survival rate, which is captured in the `outcome` variable.

Exercise 1 - proportion_treated

Since this is an RCT, the treatment column is randomized. Let's warm up by finding what the treatment probability is.

$$p_{treatment} = \frac{n_{treatment}}{n}$$

- $n_{treatment}$ is the number of patients where `TRTMT = True`
- n is the total number of patients.

In [ ]:
```
# UNQ_C1 (UNIQUE CELL IDENTIFIER, DO NOT EDIT)
def proportion_treated(df):
    """
    Compute proportion of trial participants who have been treated

    Args:
        df (dataframe): dataframe containing trial results. Column
                      'TRTMT' is 1 if patient was treated, 0 otherwise.

    Returns:
        result (float64): proportion of patients who were treated
    """

    ### START CODE HERE (REPLACE INSTANCES OF 'None' with your code) ###

    proportion = None

    ### END CODE HERE ###

    return proportion
```

In [ ]:
```
### test cell ex1: you cannot edit this cell
proportion_treated_test(proportion_treated)
```

Expected Output:

```
Test Case 1:

Example df:
```

```
    outcome  TRTMT
0        0      0
1        1      1
2        1      1
3        1      1
Proportion of patient treated: 0.75

Test Case 2:

Example df:
    outcome  TRTMT
0        0      0
1        1      0
2        0      0
3        0      0
Proportion of patient treated: 0.0

Test Case 3:

Example df:
    outcome  TRTMT
0        0      0
1        1      1
2        1      0
3        1      0
Proportion of patient treated: 0.25

 All tests passed.
```

Next let's run it on our trial data.

```
In [ ]: p = proportion_treated(data)
        print(f"Proportion Treated: {p} ~ {int(p*100)}%")
```

Exercise 2 - event_rate

Next, we can get a preliminary sense of the results by computing the empirical 5-year death probability for the treated arm versus the control arm.

The probability of dying for patients who received the treatment is:

$$p_{\text{treatment, death}} = \frac{n_{\text{treatment,death}}}{n_{\text{treatment}}}$$

- $n_{\text{treatment,death}}$ is the number of patients who received the treatment and died.
- $n_{\text{treatment}}$ is the number of patients who received treatment.

The probability of dying for patients in the control group (who did not received treatment) is:

$$p_{\text{control, death}} = \frac{n_{\text{control,death}}}{n_{\text{control}}}$$

- $n_{\text{control,death}}$ is the number of patients in the control group (did not receive the treatment) who died. - n_{control} is the number of patients in the control group (did not receive treatment).

In [ ]:
```
# UNQ_C2 (UNIQUE CELL IDENTIFIER, DO NOT EDIT)
def event_rate(df):
    '''
    Compute empirical rate of death within 5 years
    for treated and untreated groups.

    Args:
        df (dataframe): dataframe containing trial results.
                          'TRTMT' column is 1 if patient was treated, 0 otherwise.
                            'outcome' column is 1 if patient died within 5 years, 0 ot

    Returns:
        treated_prob (float64): empirical probability of death given treatment
        untreated_prob (float64): empirical probability of death given control
    '''

    treated_prob = 0.0
    control_prob = 0.0

    ### START CODE HERE (REPLACE INSTANCES OF 'None' with your code) ###

    treated_prob = None
    control_prob = None

    ### END CODE HERE ###

    return treated_prob, control_prob
```

In [ ]:
```
### test cell ex2: you cannot edit this cell
event_rate_test(event_rate)
```

Expected Output:

```
Test Case 1:

Example df:
    outcome  TRTMT
0         0      1
1         1      1
2         1      1
3         0      1
4         1      0
5         1      0
6         1      0
7         0      0

Treated 5-year death rate: 0.5
Control 5-year death rate: 0.75
```