NLP4CSS: Homework #3

Due 11:59pm EST on 03 08, 2024

Instructor: Anjalie Field; Lead TA: Samuel Lefcourt; special thanks to Carlos Aguirre

Guidelines. This assignment is to be completed **individually**. Be sure to comply with course policies on the course website.
Starter Code. Starter code is provided.

```
HW3
|- main.py
|- requirements.txt
```

Submission. This homework has written and coding components. For coding, you will complete the python file and submit it to gradescope. For the written part, you will write your answers in a PDF named `README.pdf` and also submit it to gradescope. Your PDF should contain answers to Problem 3 and Problem 5. Course Entry Code: YDPR48. Your final submission should have all the completed python file as well as your `README.pdf`.

Introduction

In class you learned about multiple aspects of causal inference. Following is a summary of key concepts:

- **Causal Inference.** Process of establishing, and quantifying causal relationships empirically
- **Treatment and Outcome.** The variables that represents possible actions (treatment) and the observed result after treatment (outcome)
- **Confounders.** Factors that are common causes of both the treatment and the outcome

In this homework, you will estimate the treatment effect in a semi-synthetic dataset based on the 20 Newsgroup dataset. The 20 Newsgroup dataset is a collection of approximately 20K newsgroup documents corresponding to different topics; some of the topics are highly related and some are not, so we relabel the topics according to subject matter in the code.
The dataset is created by the following:

$$\mathbf{Y} = const. + U\_bias * \mathbf{U} + Z\_bias * \mathbf{Z} + \epsilon, \quad (1)$$

$$\mathbf{Z} = 1.0 * U + \epsilon > 0 \quad (2)$$

Where $\mathbf{Z}$ is the treatment variable–for example, this could be the medium in which the documents where consumed: a mobile device or a desktop. $\mathbf{Y}$ is the outcome variable; using the previous example, we could estimate the effect of the medium used on the outcome of the number of lines read. And $\mathbf{U}$ is an unobserved confounder, in our example it could be the topic of the document, where we assume readers prefer to read certain topics on mobile vs desktops and the topic may also influence the number of lines in a document. Finally, ϵ is random noise.
For our homework, $\mathbf{U}$, the unobserved confounder, is whether a document belongs to a specific topic (*religion* topic is the default). The data is constructed so that $\mathbf{Z}$ is a binary variable that depends on the confounder, and $\mathbf{Y}$ depends on both $\mathbf{U}$ and $\mathbf{Z}$.
In the code, the default values for the bias values are: $const. = -0.5, U\_bias = 5.0, Z\_bias = 0.05$, and are hyperparameters to the `get_data()` function. To help your implementation, we will give you some of

the expected outputs with these parameters. We will grade submissions using different parameters. When debugging, feel free to change the values to obtain different datasets.

Problem 1: Estimating the treatment effect by regressing Y on Z only.

(10 Points) Complete the `regress_y_on_z` function. First, investigate what would happen if you don't adjust for the confounder. Complete the function to estimate the effect of z on y using the following pseudo-code:

```
def regress_y_on_z(data, max_iter):
    # format data
    # create OLS model
    res = model.fit(method=''pinv'', maxiter=max_iter)
    print(res.summary(yname=''Y'', xname=[''const'', ''Z'']))
```

You can expect the output to be similar to the following:

```
                        OLS Regression Results
==============================================================================
Dep. Variable:                      Y   R-squared:                       0.064
Model:                            OLS   Adj. R-squared:                  0.064
Method:                 Least Squares   F-statistic:                     673.1
Date:                Thu, 29 Feb 2024   Prob (F-statistic):         1.35e-143
Time:                        15:27:51   Log-Likelihood:                -18833.
No. Observations:                9816   AIC:                         3.767e+04
Df Residuals:                    9814   BIC:                         3.769e+04
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         -0.2841      0.025    -11.475      0.000      -0.333     -0.236
Z              0.8674      0.033     25.945      0.000       0.802      0.933
==============================================================================
Omnibus:                     3321.213   Durbin-Watson:                   2.014
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             8233.646
Skew:                           1.941   Prob(JB):                         0.00
Kurtosis:                       5.249   Cond. No.                         2.75
==============================================================================
```

Where the *const.* and *Z_bias* are wrongly estimated (see the `coef` column and the confidence intervals) due to confounding bias.

Problem 2: Using Confounder to Estimate Y

(10 Points) Complete the `regress_y_on_z_and_u` function. Next, assume that you do observe the confounder. Based on the `regress_y_on_z` code from the previous section, estimate the treatment effect by regressing **Y** on **Z** and the confounder **U**. You should obtain values very close to the actual bias values, however, this is an unrealistic scenario as the confounder **U** is unobserved.

Problem 3: Controlling for Confounders with Structured Text

In the previous section we had access to the confounder, but in real life scenarios this is unobserved. However, note that the confounder **U** is based on whether a document belongs to a specific topic of documents. Could we capture the confounder from observed text?

Part A (20 Points) Complete the `regress_y_on_z_and_topics` function. Instead of regressing from word statistics from all of the vocabulary tokens in the documents (it would take a long time to converge), we will instead reduce the dimensionality of texts by employing topic models. As a starting point, use a matrix factorization (NMF from sklearn with TfidfVectorizer as inputs) topic model with 50 topics. Use the NMF model to extract topics from documents and use those lower-dimensional features as well as **Z** to regress **Y**. You should find that the Z_bias estimate improves compared to Problem 1, but is less good than Problem 2.

Part B (5 Points) Find a good control. Play around with the model (e.g. try LDA instead on NMF), number of topics, other hyperparameters, or other methods (word embeddings?) to obtain a control that improves the estimate of the coefficient Z_bias. Ideally, .05 would be within the confidence intervals, but you do not have to achieve this for full credit. In your written report, write a short paragraph describing what you tried and what the results were. Please submit your code so that the 50-topic NMF model described above runs by default (You can put your experimental code in a different function or use an optional parameter to control what model runs).

Problem 4: Controlling for Confounders with Inverse Probability Weighting

In class you learned about inverse probability weighting, which assigns different weights to subjects based on their propensity scores to account for confounders

(20 Points) Complete the `reweigh_with_propensity_scores` function. As in Problem 3, we will assume we do not directly know the confounder, and we will use the text as a proxy. Obtain propensity scores by training a logistic regression model using TF-IDF-weighted features (we suggest sklearn with max_iter=2000 for the logistic regression model). To avoid overfitting, split the data into 2 equal-sized halves: *train* (used to train the propensity score model) and *test* (you can use `train_test_split` from sklearn). Use the propensity scores to reweigh **Y**, and return the both the unadjusted and the adjusted average treatment effect estimated over the test set. (Hint: Adjusted=0.580, Unadjusted=0.866)

Problem 5: Draw Causal Graphs

For the following brief scenarios where text may be employed to help examine causal relations, identify the treatment and the outcome(s) and potential confounder(s) and briefly justify your answer. Speculate what may be the possible confounders if none are evident from the description. Then, draw the causal graph. Your answer should be in the form:

Treatment: *your answer*
Outcome(s): *your answer*
Confounder(s): *your answer*
Your graph, using any software or hand draw-and-scan you prefer as long as we can read it

1. From the population of users tweeting about mental health, what linguistic structures or linguistic patterns differentiates those who proceed to discuss suicidal ideation in the future, from those who do not?

Problem 5 continued on next page...

2. By collecting Reddit timelines from students entering college, we study what effect does drinking early in college have on college success, including habits, social relationships, and even criminal activity, of those who mention drinking during their first semester versus those that do not.

3. We collect politically charged debates and investigate how the tones of the statements made during the debates changes the linguistic and sentiment characteristics in subsequent responses.

4. If an AI article published under a woman's name were instead published in the same venue under the name of a man with the same scholarly credentials, would it be cited more?