Overview

Submit your writeup including all code and plots as a PDF via Gradescope.[1] We recommend reading through the entire homework beforehand and carefully using functions for testing procedures, plotting, and running experiments. Taking the time to test, maintain, and reuse code will help in the long run!

Data science is a collaborative activity. While you may talk with others about the homework, please write up your solutions individually. If you discuss the homework with your peers, please include their names on your submission. Please make sure any handwritten answers are legible, as we may deduct points otherwise.

Please note that this homework is slightly shorter than usual, to give you time to start working on your project.

1 Observational Data on Infant Health

The Infant Health and Development Program (IHDP) was an experiment treating low-birth-weight, premature infants with intensive high-quality childcare from a trained provider. The goal is to estimate the causal effect of this treatment on the child's cognitive test scores. The data *does not* represent a randomized trial with randomly allocated treatment, so there may be confounders between treatment and outcome. In this problem, we devise a propensity score model to control for observed confounders.

(a) (2 points) The CSV file `ihdp.csv` has 27 columns:

- Column 1 is the treatment $z_i \in \{0, 1\}$, which indicates whether or not the treatment was given to the infant.
- Column 2 is the outcome $y_i \in \mathbb{R}$, the child's cognitive test score.
- Columns 3-27 contain 25 features of the mother and child (*e.g.* the child's birth weight, whether or not the mother smoked during pregnancy, her age and race). Since this dataset was not collected by a randomized trial, these features could all confound z_i and y_i, and are denoted by $x_i \in \mathbb{R}^{25}$.

In this part, you'll estimate $\hat{e}(x)$ (the predicted probability that $z_i = 1$) by fitting a logistic regression model that predicts z_i from x_i. Specifically:

1. Read the data in `ihdp.csv` (*e.g.* using the `csv` package in Python) into three arrays: $Z \in \{0, 1\}^n$ containing the treatments, $Y \in \mathbb{R}^n$ containing the outcomes, and $X \in \mathbb{R}^{n \times 25}$ containing the features.
2. To fit a logistic regression model, use the `scikit-learn` package in Python, which is imported as `sklearn`. Start with the following two lines:

[1] In Jupyter, you can download as PDF or print to save as PDF

```
from sklearn.linear_model import LogisticRegression as LR
lr = LR(penalty='none', max_iter=200, random_state=0)
```

3. Use the `lr.fit()` method to fit the logistic regression model $\hat{e}(x)$

See the documentation here.

(b) (2 points) Write a function `estimate_treatment_effect` to estimate treatment accounting for the propensity. It should take as arguments a fitted regression model (the `LogisticRegression` object `lr` from the previous part), X, Y, and Z, and output a single value, which is the estimate of the average treatment effect.

Hint: Use the inverse propensity weighted estimator:

$$\hat{\tau} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{z_i y_i}{\hat{e}(x_i)} - \frac{(1-z_i)y_i}{1-\hat{e}(x_i)}\right). \tag{1}$$

See the `LogisticRegression` object's `predict_proba` method.

(c) (3 points) Use the function `estimate_treatment_effect` from the previous part to estimate the treatment effect on the IHDP dataset. Report this estimate. According to the estimate, did the treatment have a beneficial causal effect on the outcome (*i.e.* cause cognitive test scores to increase)?

(d) (3 points) The naïve estimator is the difference between the sample means:

$$\tilde{\tau} = \frac{1}{n_1}\sum_{i=1}^{n} y_i z_i - \frac{1}{n_0}\sum_{i=1}^{n} y_i(1-z_i), \tag{2}$$

where $n_1 = \sum_{i=1}^{n} z_i$ and $n_0 = n - n_1$. Report this estimate on the IHDP dataset. Why is it different from the estimate you computed in the previous part? Are there any circumstances under which these two estimators should produce the same estimates?

2 Causal Inference Potpourri

A research team wants to estimate the effectiveness of a new veterinary drug for sick seals. They ask aquariums across the country to volunteer their sick seals for the experiment. Since the team offers monetary compensation for volunteering, zoos with less income decide to volunteer their sick seals, whereas zoos with more income are less compelled to volunteer their seals.

It turns out that zoos with less income feed their seals less nutritious diets (regardless of whether they are sick or healthy), due to budgetary constraints. Less nutritious diets prevent seals from recovering as effectively.

(a) (2 points) **Draw** a causal graph between variables X, Y, I and N which denote receiving the drug, recovering, the income level of the zoo, and how nutritious a seal's diet is, respectively. Justify each edge in your graph.

(b) (3 points) We saw in lecture that if we can identify and condition on (adjust for) all confounding variables, then we can use the unconfoundedness assumption to compute the average treatment effect (ATE).

The *backdoor criterion* provides a way to determine which variables are confounders. In particular, we simply need to "block" all the confounding pathways in the graphical model between X and Y.

In a causal graph, we define a *path* between two nodes X and Y as a sequence of nodes beginning with X and ending with Y, where each node is connected to the next by an edge (pointed in either direction).

Given an ordered pair of variables (X, Y), a set of variables S satisfies the backdoor criterion relative to (X, Y) if no node in S is a descendant of X (to prevent us from conditioning on colliders), and S blocks every path between X and Y that contains an arrow into X.

Using the causal graph in the previous part, **determine all possible sets of variables** that satisfy the backdoor criterion relative to (X, Y).

(c) (3 points) Read the following paper: *Safety of the BNT162b2 mRNA Covid-19 Vaccine in a Nationwide Setting* and answer the following questions. As when reading any academic paper, you don't have to understand every single thing in the paper as you read: focus on connecting what you're reading to what we learned in class.

(i) Does the paper use one of the causal inference techniques we learned in class? If so, which one? Do you have any concerns with how it was applied?

(ii) The paper answers two distinct causal questions. For each of the two questions, what is the causal question? What is the treatment? What is the outcome(s)? What are the confounding variables? If any instrumental variables are used, what are they?

(iii) Identify at least one potential concern or limitation with this study.
Hint: when reading any paper, the discussion and/or conclusion sections are often the most insightful.

3 California House Values

In this problem, we will apply three methods (linear regression, decision trees, and random forests) to predict the median value of houses per census block in California. The data we will use is contained in the CSV file `housing.csv` and was obtained from the 1990 census. Each row provides information about one census block.

(a) (3 points) Read the data in `housing.csv`. The dataset has 10 columns, and we will focus on a subset of them: our outcome y will be the log of `median_house_value`, and the features X we will use are `latitude`, `longitude`, `housing_median_age`, `median_income`, `households`, and `ocean_proximity`.

1. Log-transform the outcome variable.
2. Select the features we are interested in (described above), and note that `ocean_proximity` is the only non-numeric one. Use one-hot-encoding to transform each text label into a binary column.
   Hint: you can use `pd.get_dummies` or `OneHotEncoder` from `scikit-learn`.
3. Split the data into training (70%) and test (30%) set.
4. Use `scikit-learn` to fit the following models: linear regression, decision trees, and random forests.
5. For each model, compute the mean squared error (MSE) on the training and test sets.

In your solution, please include your code, and the MSEs obtained in point 5.

(b) (2 points) Interpret the results you obtained. For each method, describe how the median house value changes if the longitude increases by one unit, or explain why this can't be done using what we learned in class.

Question 1

a) **Part a**

```
ihdp = pd.read_csv('ihdp.csv')
```

```
from sklearn.linear_model import LogisticRegression as LR
lr = LR(penalty='none', max_iter=200, random_state=0)
```

```
Z = ihdp.iloc[:, 0]
```

```
Y = ihdp.iloc[:, 1]
```

```
X = ihdp.iloc[:, 2:]
```

```
lr.fit(X, Z)
```

```
LogisticRegression(max_iter=200, penalty='none', random_state=0)
```

b) **Part b**

```
def estimate_treatment_effect(lr, X, Y, Z):
    e_x = lr.predict_proba(X)[:, 1]
    return np.mean(Z*Y/e_x) - np.mean(((1-Z)*Y)/(1-e_x))
```

c) **Part c**

```
treatment_effect = estimate_treatment_effect(lr, X, Y, Z)
treatment_effect
```

```
3.6956988430254074
```

According to the calculation above, the treatment effect is around 3.7. Since the treatment effect is positive, this illustrates that the treatment has a beneficial effect on cognitive test scores.

d) **Part d**

```
y_treatment = ihdp[ihdp['treatment'] == 1]['outcome']
y_control = ihdp[ihdp['treatment'] == 0]['outcome']
```

```
z_treatment = ihdp[ihdp['treatment'] == 1]['treatment']
z_control = ihdp[ihdp['treatment'] == 0]['treatment']
```

```
np.mean(y_treatment*z_treatment) - np.mean(y_control*(1-z_control))
```

```
4.021121012430829
```

This estimate is roughly 4.0.

The reason it's different from the previous part is because this part does not take into account any x. What this means is that we are ignoring the existence of confounders (X) between Z and Y. By ignoring the confounding variables, the causal effect in part(d) is bigger than the causal effect in part(c).

In order for these two estimators to produce the same estimates, we need:

$$\hat{\tau} = \tilde{\tau}$$

$$\frac{1}{n}\sum_{i=1}^{n}\left(\frac{z_i y_i}{\hat{e}(x_i)} - \frac{(1-z_i)y_i}{1-\hat{e}(x_i)}\right) = \frac{1}{n_1}\sum_{i=1}^{n} y_i z_i - \frac{1}{n_0}\sum_{i=1}^{n} y_i(1-z_i)$$

$$\frac{1}{n}\sum_{i=1}^{n}\frac{z_i y_i}{\hat{e}(x_i)} - \frac{1}{n}\sum_{i=1}^{n}\frac{(1-z_i)y_i}{1-\hat{e}(x_i)} = \frac{1}{n_1}\sum_{i=1}^{n} y_i z_i - \frac{1}{n_0}\sum_{i=1}^{n} y_i(1-z_i)$$

$$\frac{1}{\hat{e}(x_i).n}\sum_{i=1}^{n} z_i y_i - \frac{1}{(1-\hat{e}(x_i))n}\sum_{i=1}^{n}(1-z_i)y_i = \frac{1}{n_1}\sum_{i=1}^{n} y_i z_i - \frac{1}{n_0}\sum_{i=1}^{n} y_i(1-z_i)$$

$$\hat{e}(x_i).n = n_1 \Rightarrow \hat{e}(x) = \frac{n_1}{n}$$

$$(1-\hat{e}(x_i))n = n_0 \Rightarrow n - n\hat{e}(x_i) = n_0$$

$$n - n_0 = n\hat{e}(x_i)$$

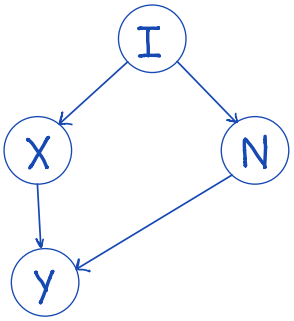
$$n_1 = n\hat{e}(x_i)$$

$$\hat{e}(x_i) = \frac{n_1}{n}$$

Hence the condition under which $\hat{\tau} = \tilde{\tau}$ is if $\hat{e}(x_i) = \frac{n_1}{n}$.

QUESTION 2

a)



X = receiving the drug

Y = recovering

I = income level of the zoo

N = how nutritious a seal's diet is

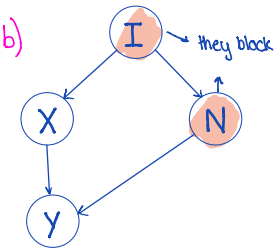
I → N: the higher the income of the zoo, the more nutritious the seal's diet is

I → X: the higher the income of the zoo, the more likely the seal is to receive the drug → lower income volunteers seals

X → Y: if a seal receives the drug, it is more likely to recover.

N → Y: the more nutritious a seal's diet is, the more likely it is to recover

b)



Backdoor criteria w.r.t (X,Y)

1. No $s \in S$ is a descendent of X $\Rightarrow Y \notin S$
2. S blocks every path between X, Y that with $\rightarrow X \Rightarrow I \in S, N \in S$

Possible sets w I, N: $\boxed{\{I,N\}, \{I\}, \{N\}}$

c) i.

Part c (i)

Does the paper use one of the causal inference techniques we learned in class? If so, which one? Do you have any concerns with how it was applied?

Yes, this paper uses the matching technique.

My concern with how the matching process was applied was that it resulted in a study population with a different composition than the eligible population (e.g., median age, 38 years rather than 43 years). The study also excluded certain populations (such as health care workers and persons residing in long-term care facilities) that could be at particularly high risk for certain adverse events. These two factors affected the demographic composition of the study, and are a cause of concern because they affect the generalizability of the study.

c) ii.

Part c (ii)

The paper answers two distinct causal questions. For each of the two questions: What is the causal question? What is the treatment? What is the outcome(s)? What are the confounding variables? If any instrumental variables are used, what are they?

Causal question 1:

Question: What is the causal effect of the COVID mRNA vaccine on adverse events?

Treatment: Receiving the COVID mRNA vaccine

Outcome: Experiencing adverse health events or not

Confounding variables: Sociodemographic variables: age, sex, place of residence, socioeconomic status, and population sector. Clinical variables: number of preexisting chronic conditions, number of diagnoses documented in outpatient visits in the year before the index date, and pregnancy status.

Instrumental variables: None

Causal question 2:

Question: What is the causal effect of being infected on adverse events?

Treatment: Being infected with COVID

Outcome: Experiencing adverse health events or not

Confounding variables: Sociodemographic variables: age, sex, place of residence, socioeconomic status, and population sector. Clinical variables: number of preexisting chronic conditions, number of diagnoses documented in outpatient visits in the year before the index date, and pregnancy status.

Instrumental variables: None

c) iii.

Part c (iii)

Identify at least one potential concern or limitation with this study.

One potential limitation with this study was that persons in the study were not randomly assigned according to exposures. This may have introduced confounding at baseline and selection bias at censoring, which affects the generalizability of the study.

QUESTION 3

a) **Part a**

```
housing = pd.read_csv('housing.csv')
```

```
housing['log_median_house_value'] = np.log(housing['median_house_value'])
```

```
housing = pd.get_dummies(housing, columns=['ocean_proximity'])
```

```
housing.columns
```

```
Index(['longitude', 'latitude', 'housing_median_age', 'total_rooms',
       'total_bedrooms', 'population', 'households', 'median_income',
       'median_house_value', 'log_median_house_value',
       'ocean_proximity_<1H OCEAN', 'ocean_proximity_INLAND',
       'ocean_proximity_ISLAND', 'ocean_proximity_NEAR BAY',
       'ocean_proximity_NEAR OCEAN'],
      dtype='object')
```

```
y = housing['log_median_house_value']
X = housing[['latitude', 'longitude', 'housing_median_age', 'median_income', 'households',
             'ocean_proximity_<1H OCEAN', 'ocean_proximity_INLAND', 'ocean_proximity_ISLAND',
            'ocean_proximity_NEAR BAY', 'ocean_proximity_NEAR OCEAN']]
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)
```

```
from sklearn.linear_model import LinearRegression
linear_model = LinearRegression().fit(X_train, y_train)
```

```
from sklearn.tree import DecisionTreeRegressor
decision_tree_model = DecisionTreeRegressor().fit(X_train, y_train)
```

```
from sklearn.ensemble import RandomForestRegressor
random_forest_model = RandomForestRegressor().fit(X_train, y_train)
```

```
print('linear model training mse: ', np.mean((y_train - linear_model.predict(X_train))**2))
print('linear model test mse: ', np.mean((y_test - linear_model.predict(X_test))**2))
```

```
linear model training mse:  0.11586334276595661
linear model test mse:  0.11862974363477974
```

```
print('decision tree model training mse: ', np.mean((y_train - decision_tree_model.predict(X_train))**2))
print('decision tree model test mse: ', np.mean((y_test - decision_tree_model.predict(X_test))**2))
```

```
decision tree model training mse:  1.0352161380807298e-31
decision tree model test mse:  0.09953818484254016
```

```
print('random forest model training mse: ', np.mean((y_train - random_forest_model.predict(X_train))**2))
print('random forest model test mse: ', np.mean((y_test - random_forest_model.predict(X_test))**2))
```

```
random forest model training mse:  0.007360816511787933
random forest model test mse:  0.05711941827263565
```

Steps I took: Read in the dataset, log-transformed the outcome variable, one-hot-encoded 'ocean_proximity' since it was a non-numeric variable, performed the train test split, fit the models, calculated the MSEs.

b) For the MSE of the decision tree on the training set, it makes sense that it is very, very small (6.53×10^{-32}) because decision trees tend to gain 100% accuracy on the training set by predicting all points correctly. They do this by overfitting the training data which leads to high MSE on the test set because there is high variance. The random forest model fixes this problem by averaging on a lot of decision trees, so the MSE of the test set is lowered. However, the tradeoff here is that the training set MSE for this model is higher than the training set MSE for the decision tree model.

Lastly, for the linear regression model, the MSEs are higher than the random forest and decision tree models. The training set and test set MSEs are also very, very similar.

```
linear_model.coef_
```

```
array([-1.42282981e-01, -1.47051950e-01,  2.78596764e-03,  1.66741340e-01,
        1.38436098e-04, -7.98993420e-02, -3.99987555e-01,  6.48670450e-01,
       -8.04126446e-02, -8.83709082e-02])
```

```
np.exp(linear_model.coef_[1])
```

```
0.863249130863866
```

For the linear model, estimating the change in median house value if longitude increases by one unit is equivalent to a -1.51×10^{-1} increase in the log of median house value and a $\exp(-1.51 \times 10^{-1}) = 0.86$ increase in median house value

For the decision tree and random forest model, there is no quick way of interpreting the effect of a one unit change in longitude because these models do not have coefficients, and hence we cannot draw these conclusions in the same way.